L Number	Hits	Search Text	DB	Time stamp
1	1197	remote near10 backup	USPAT; US-PGPUB	2004/06/09 09:53
2	341	network same (remote near10 backup)	USPAT; US-PGPUB	2004/06/09 09:54
3	24	(message or token) same (network same (remote near10 backup))	USPAT; US-PGPUB	2004/06/09 10:12
4	1618	local near10 server same central near10 server	USPAT; US-PGPUB	2004/06/09 16:28
5	187	(token or message) same (local near10 server same central near10 server)	USPAT; US-PGPUB	2004/06/09 10:14
6	75	(access or request) same ((token or message) same (local near10 server same central near10 server))	USPAT; US-PGPUB	2004/06/09 10:19
7	51	(synchroniz$ or coheren$) same (local near10 server same central near10 server)	USPAT; US-PGPUB	2004/06/09 10:19
10	1	10/686318	USPAT; US-PGPUB	2004/06/09 14:22
27	4547	server near10 cache	USPAT; US-PGPUB	2004/06/09 16:26
28	207	(server near10 cache) near10 (coheren$ or synchroniz$)	USPAT; US-PGPUB	2004/06/09 16:27
29	1	(local near10 access) near10 ((server near10 cache) near10 (coheren$ or synchroniz$))	USPAT; US-PGPUB	2004/06/09 16:27
30	6	(local near10 access) same ((server near10 cache) near10 (coheren$ or synchroniz$))	USPAT; US-PGPUB	2004/06/09 16:27
31	1083	local near5 server same central near5 server	USPAT; US-PGPUB	2004/06/09 16:28
32	33	(local near5 server same central near5 server) same cache	USPAT; US-PGPUB	2004/06/09 16:28

US005802297A

United States Patent [19]

Engquist

[11] Patent Number: 5,802,297

[45] Date of Patent: Sep. 1, 1998

[54] **CLIENT-SERVER COMPUTER SYSTEM AND METHOD UTILIZING A LOCAL CLIENT DISK DRIVE AS A DATA CACHE**

[75] Inventor: **James D. Engquist**, Colorado Springs, Colo.

[73] Assignee: **Sun Microsystems, Inc.**, Palo Alto, Calif.

[21] Appl. No.: **497,290**

[22] Filed: **Jul. 3, 1995**

[51] **Int. Cl.**[6] **G06F 12/00**

[52] **U.S. Cl.** **395/200.42**; 395/200.52

[58] **Field of Search** 395/200.01, 200.54, 395/200.42, 200.33, 200.52

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,887,204	12/1989	Johnson	395/200
4,897,781	1/1990	Chang	395/200
5,146,568	9/1992	Flaherty	395/325
5,162,986	11/1992	Graber	364/146
5,218,695	6/1993	Noveck	395/600
5,230,052	7/1993	Dayan	395/700
5,355,453	10/1994	Row	395/200
5,452,454	9/1995	Basu	395/700

FOREIGN PATENT DOCUMENTS

0 572 332 A1 12/1993 European Pat. Off. .

OTHER PUBLICATIONS

Smith, Alan Jay, "Cache Memories," *Computing Surveys*, vol. 14, No. 3, Sep. 1982, pp. 473–530.

"Cache File System (CacheFS) White Paper," SunSoft, Sun Microsystems, Inc., Revision A, Feb., 1994, Mountain View, CA, pp. 1–11.

Article Entitled "Andrew: A Distributed Personal Computing Environment" by J. H. Morris et al, Communications Of The ACM, Mar. 1986, vol. 29, No. 3 pp. 184–201.

IBM Technical Disclosure Bulletin, vol. 35, No. 1A, Jun. 1992, Entitled "AFS Initialization Performance Enhancement" pp. 217–218.

IBM Technical Disclosure Bulletin, vol. 36, No. 3, Mar. 1993, Entitled "Local Cache for Server Files", pp. 541–543.

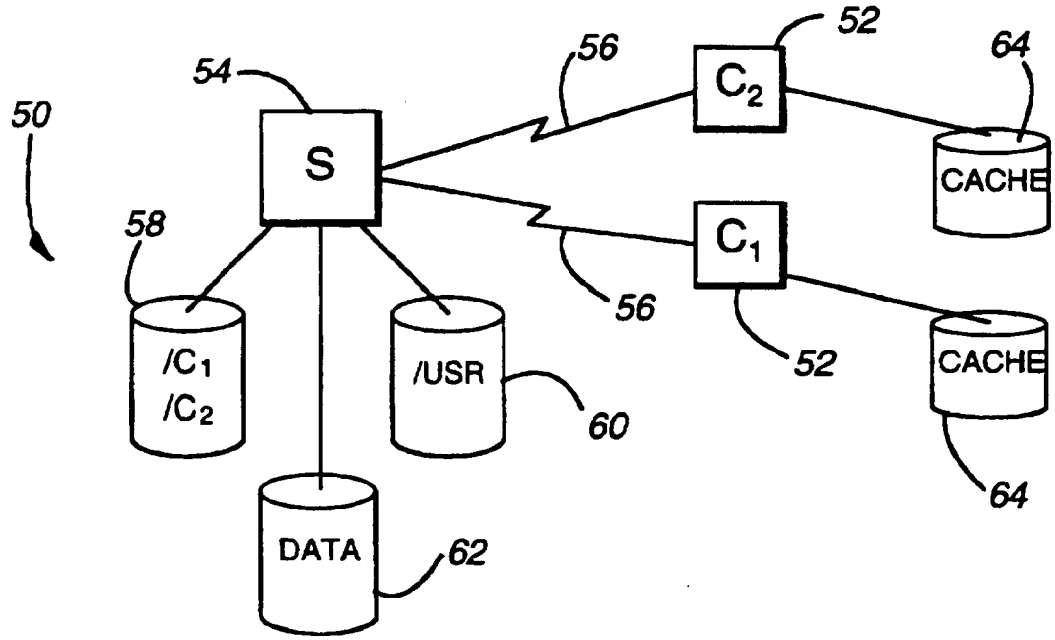
Primary Examiner—Eric Coleman

Attorney, Agent, or Firm—William J. Kubida; Stuart T. Langley; Holland & Hart LLP

[57] **ABSTRACT**

A cache only client-server configuration which provides the performance benefits of "dataless" client operation with the administrative efficiencies of a "diskless" client-server configuration. Utilizing cache only clients, the performance of stand-alone systems can be approximated utilizing a relatively small disk drive as a local data cache. The cache only clients may be considered as interchangeable units in that they hold no critical data and any data held on the local disk is a "clone" of the master copy held on the server. System configuration, administration and maintenance costs are dramatically reduced since software installation, distribution and backup may be managed at the server.

28 Claims, 5 Drawing Sheets



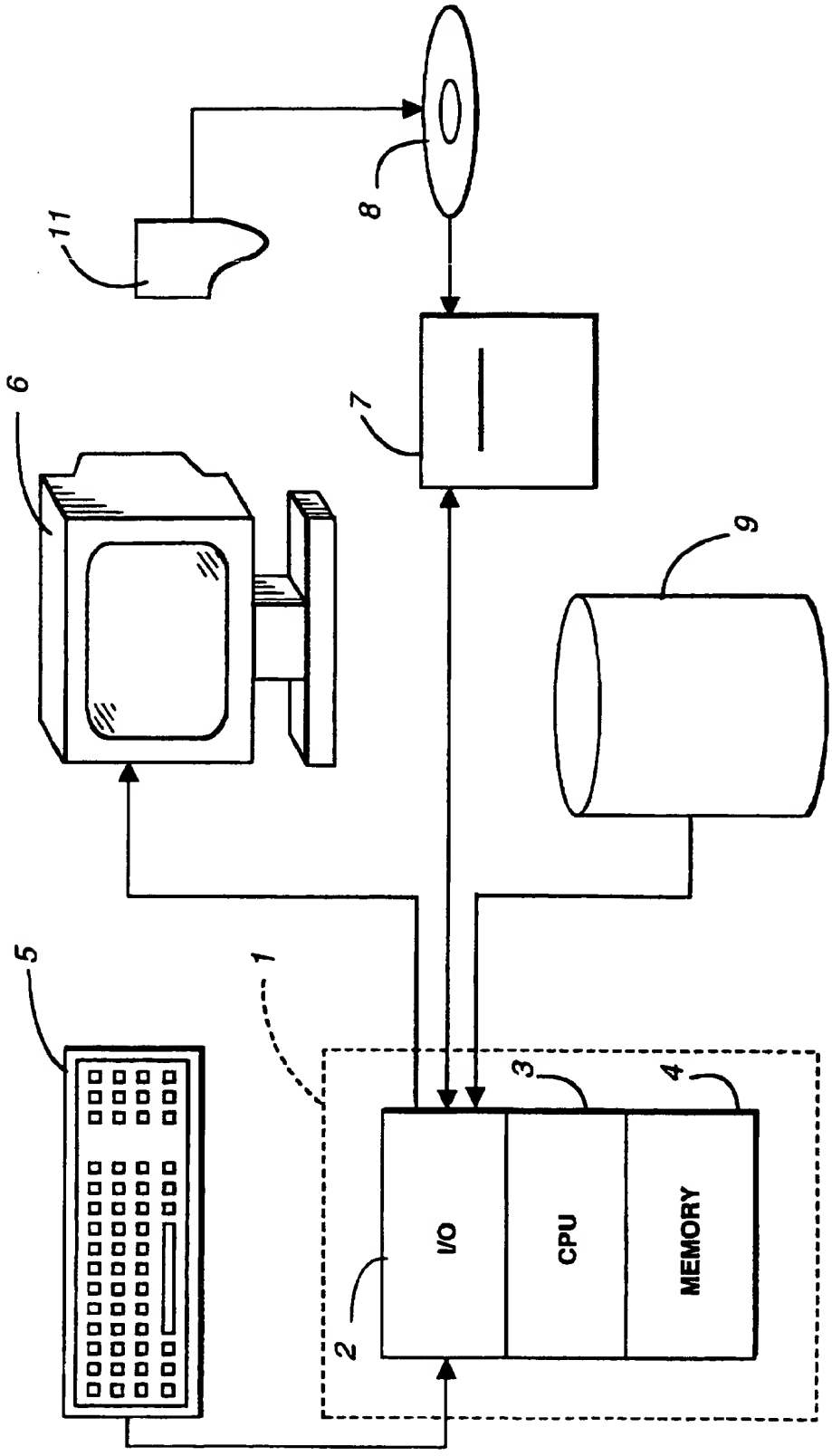


Fig. 1A

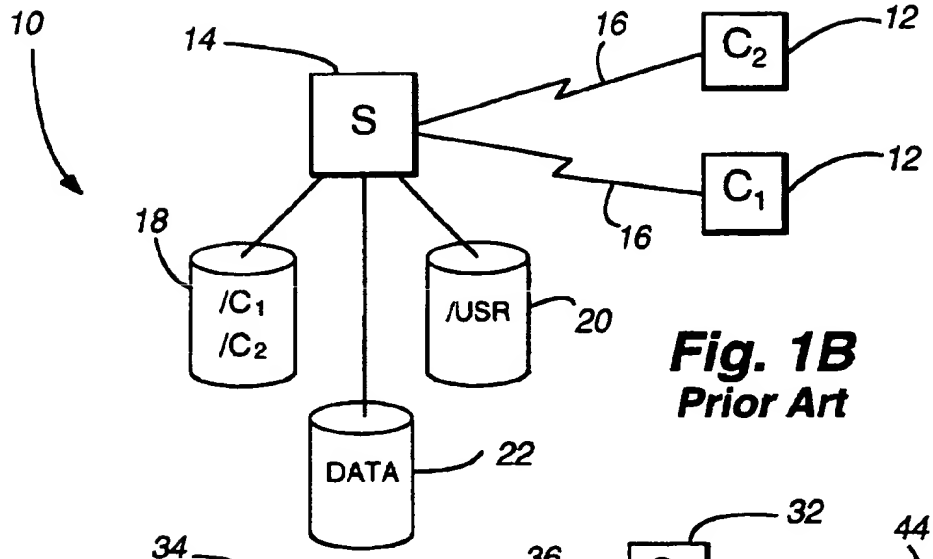


Fig. 1B
Prior Art

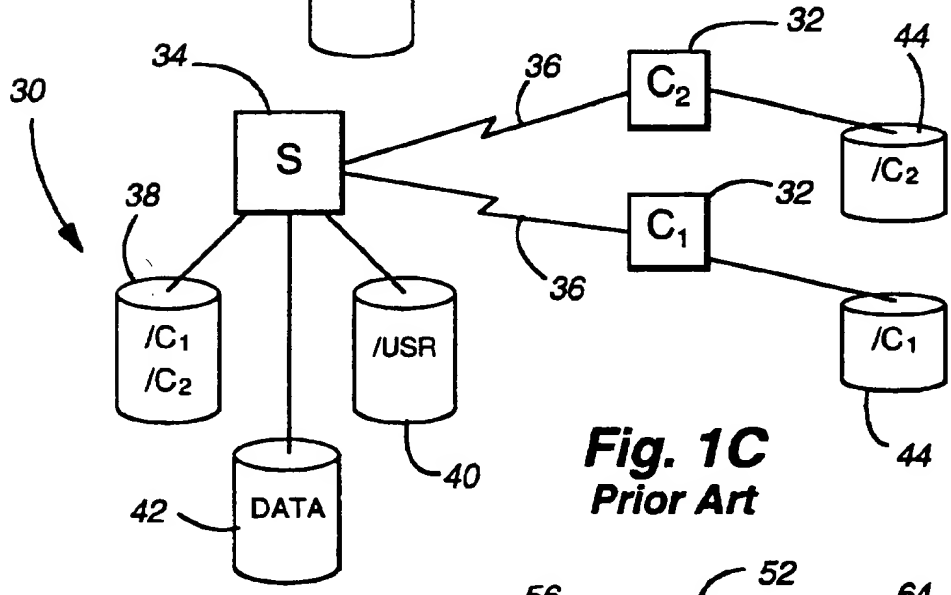


Fig. 1C
Prior Art

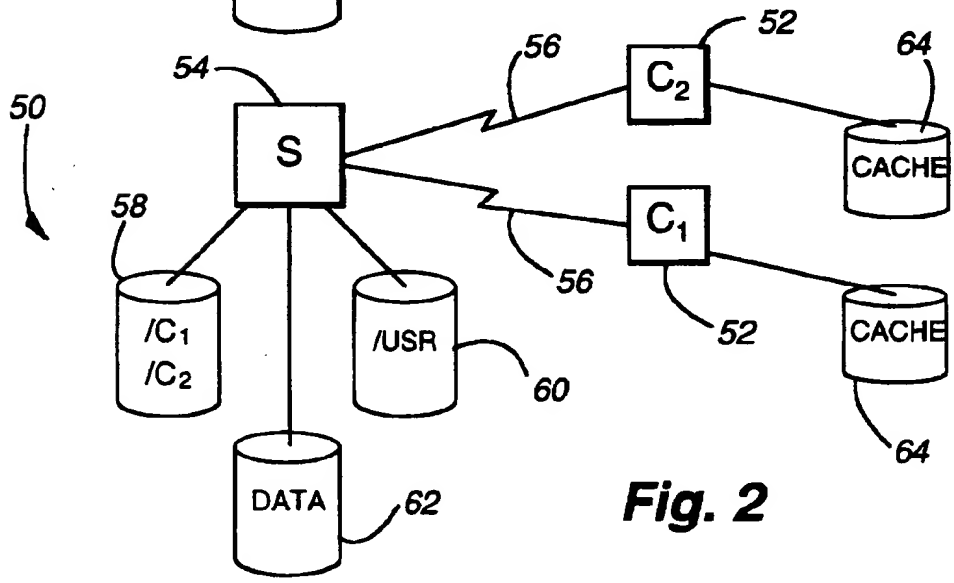


Fig. 2

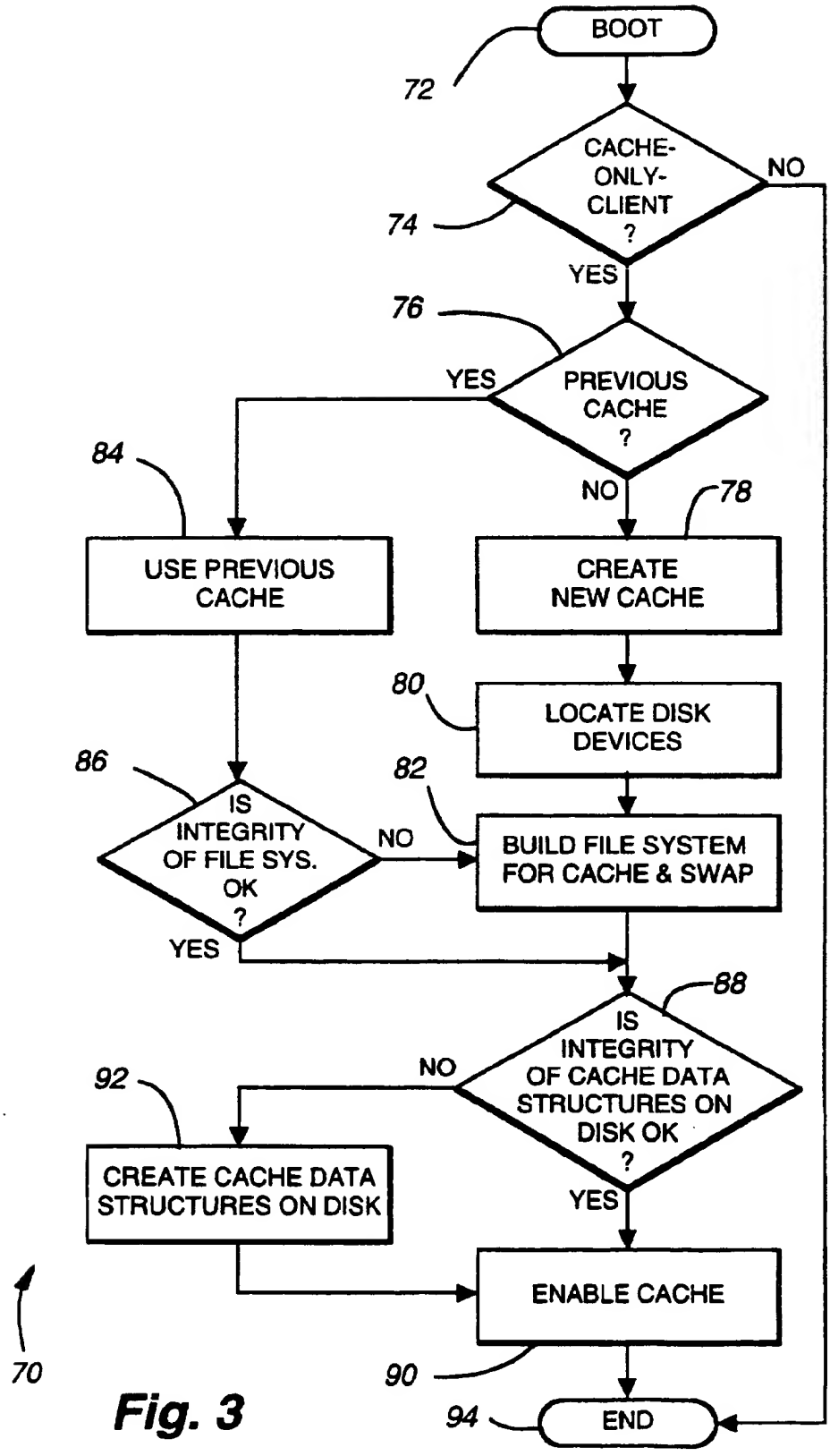

BOOT
72
CACHE-ONLY-CLIENT ?
74
NO
YES
76
PREVIOUS CACHE ?
YES
NO
84
78
USE PREVIOUS CACHE
CREATE NEW CACHE
80
LOCATE DISK DEVICES
86
82
IS INTEGRITY OF FILE SYS. OK ?
NO
BUILD FILE SYSTEM FOR CACHE & SWAP
YES
88
IS INTEGRITY OF CACHE DATA STRUCTURES ON DISK OK ?
NO
92
YES
CREATE CACHE DATA STRUCTURES ON DISK
ENABLE CACHE
70
90
94
END

Fig. 3

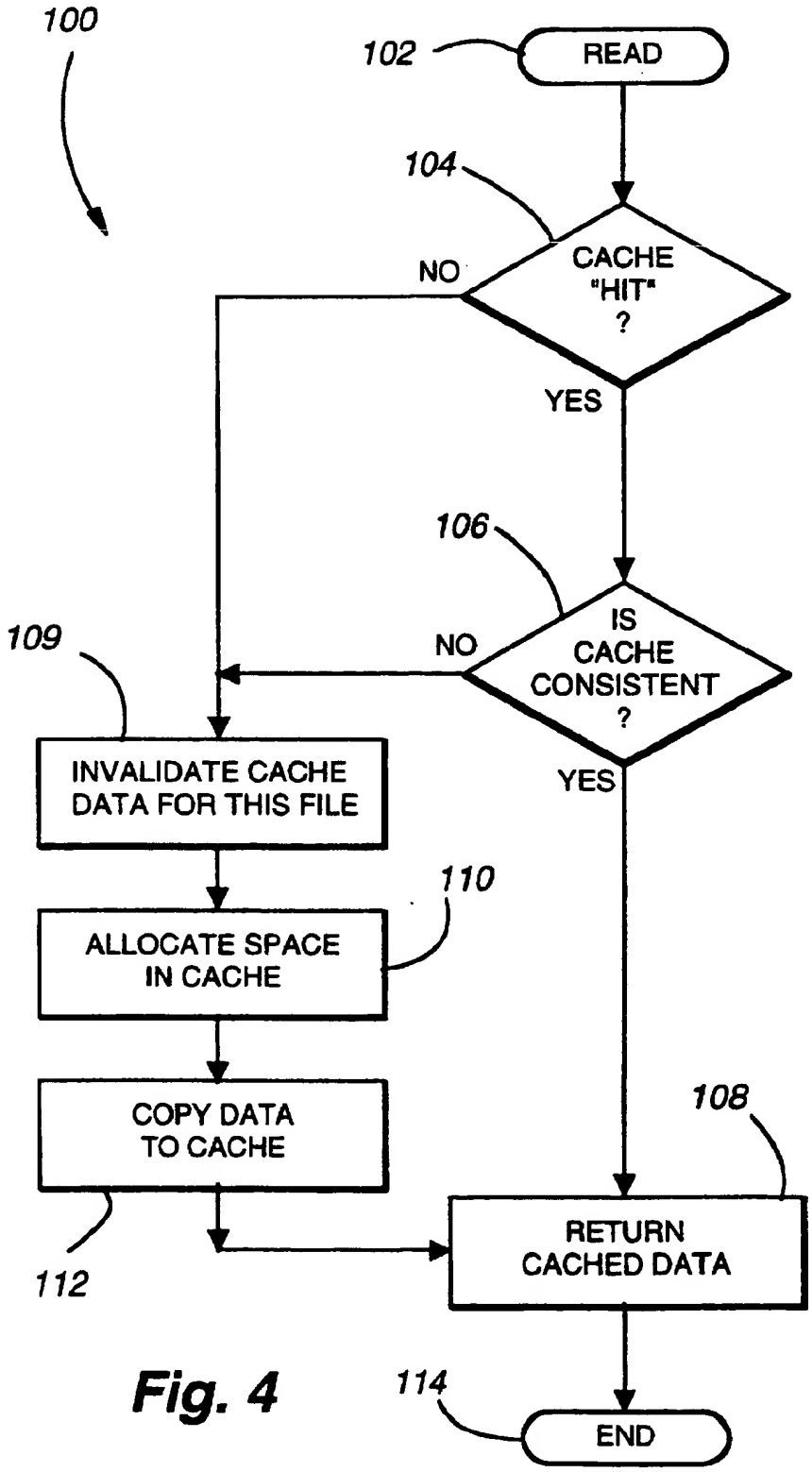

100
102
READ
104
CACHE "HIT" ?
NO
YES
106
IS CACHE CONSISTENT ?
NO
YES
109
INVALIDATE CACHE DATA FOR THIS FILE
110
ALLOCATE SPACE IN CACHE
COPY DATA TO CACHE
112
108
RETURN CACHED DATA
114
END

Fig. 4

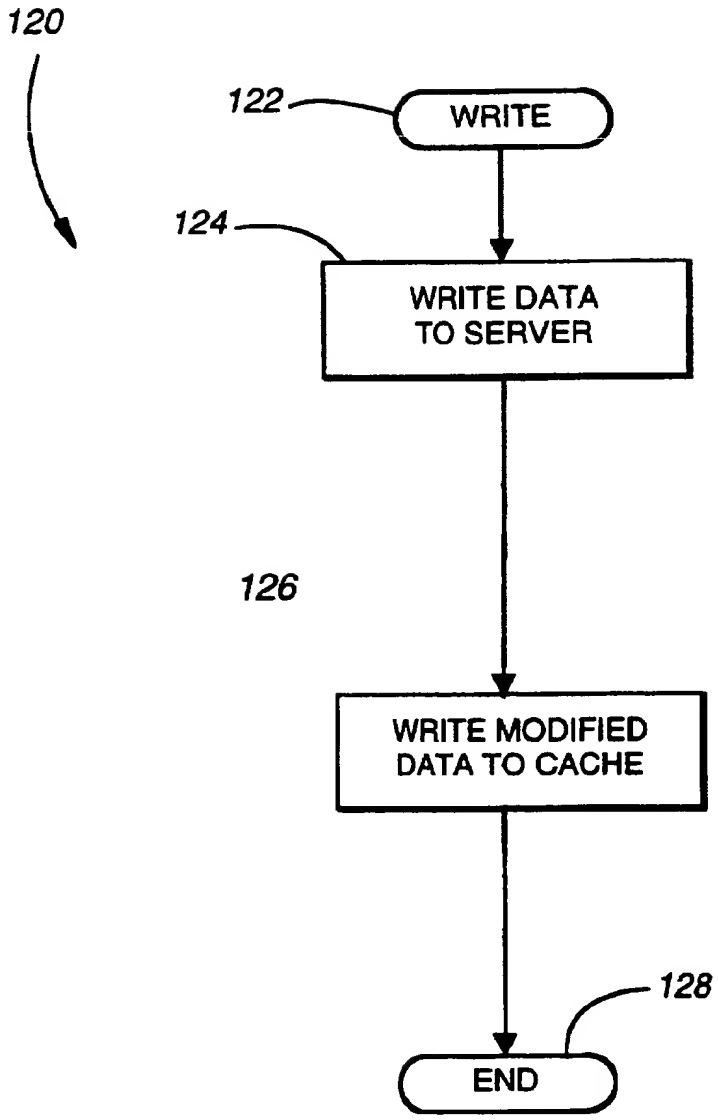

120
122
WRITE
124
WRITE DATA TO SERVER
126
WRITE MODIFIED DATA TO CACHE
128
END

Fig. 5

CLIENT-SERVER COMPUTER SYSTEM AND METHOD UTILIZING A LOCAL CLIENT DISK DRIVE AS A DATA CACHE

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of networked computer systems configured in a client-server relationship. More particularly, the present invention relates to a client-server computer system and method for operating the same utilizing a local disk drive at one or more client computers as a data cache and swap space for the root ("/") and user ("/usr") files associated with the client computer normally held in server computer mass storage.

Management information system ("MIS") managers have long been faced with the fundamental dilemma of administering a large number of distributed client computers while simultaneously maintaining network resources and their performance at acceptable levels. In conjunction with the objective of lowering the overall cost of managing such a distributed computing environment, system administrators have long recognized the need for techniques that would allow them to manage more clients per server as well as to replicate system support for a growing number of users.

In the past, centralization of system administration was used to address some of these aims. However, centralized system administration of prior client-server configurations has generally caused system performance trade-offs that were not acceptable. Thus, it has long been highly desirable to find a way to combine the benefits of centralized system administration with the higher performance demanded of current distributed computing resources. This is particularly the case with respect to system administration of UNIX® systems, where the need for centralization of desktop data is particularly acute.

Conventional approaches to client-server configurations that enable data centralization, such as "diskless" and "dataless" clients, frequently overwhelm networks and their associated servers with network "traffic" congestion. In these configurations, as additional client computers are added to a network file system, the load on the server and network tends to increase linearly. This is the primary reason servers and networks are so quickly saturated in a diskless client configuration. When servers and/or networks are overloaded the performance of all clients suffers and every time a new client is added to the configuration, the performance of all existing clients is even further degraded.

As a result, and in order to avoid these network overload problems, many sites install desktop computers as stand-alone machines. A stand-alone computer has its operating system installed on a local disk drive and, as the name implies, there is no reliance on a centralized server. As a result, the typical stand-alone desktop computer requires large amounts of disk space for the installation of operating system ("OS") components that may never be used. Although the performance of a stand-alone computer is very good and there are no scalability problems with completely independent units having no reliance on a server, because the OS must be individually loaded to each desktop computer and the data individually backed up at each location, this configuration is very difficult to administer.

SUMMARY OF THE INVENTION

Disclosed herein is a system and method for implementing a client-server computer system in which the local client disk drive is utilized as a data cache in a novel client-server workstation configuration that preserves the system administration advantages of traditional client-server configurations (both diskless and dataless) while addressing the shortcomings previously noted. The system and method of the present invention maintains data centralization and optimizes disk and network utilization by caching data that the client actually utilizes on a frequent basis. This substantially reduces demands otherwise placed on the network.

The introduction of a local disk cache on each client computer in accordance with the present invention, also reduces the client's demands on the server with a corresponding reduction in network and server load. A client can retrieve accessed files from a populated local disk drive cache with no server interaction and no additional server or network load.

Additionally, the system and method of the present invention provides a number of system management advantages over conventional approaches and installation of a client computer utilizing the local disk drive as a data cache is much quicker and simpler than a traditional stand-alone installation. All client data is centralized on the server obviating desktop backups. Moreover, since the data is centralized and loaded on-demand, individual client computer failures may be addressed by a relatively simple hardware replacement and network reboot. Still further, additional client computers may be added to a configuration in "batch" mode reducing an administrator's effort and the level of skill required as well as minimizing the wait time between each client installed.

Centralization of client data utilizing the system and method herein disclosed provides other important benefits to the system administrator in that client computers need not be individually backed up and may be treated as field replaceable units ("FRUs"). Further, in trouble shooting situations it is easy to view and modify client data from the server as well as to develop shell scripts that iteratively apply modifications to all clients.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent, and the invention itself will be best understood, by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

FIG. 1A is a simplified representational drawing of a general purpose workstation computer forming a portion of the operating environment of the present invention;

FIG. 1B is a simplified block diagram of a conventional "diskless" client-server computer configuration wherein the server maintains the root, common user and swap files for a given client computer and all data remotely from the networked client computers;

FIG. 1C is an additional, simplified block diagram of a conventional "dataless" client-server computer configuration wherein the server maintains common user files for a given client computer and all data remotely from the networked client computers while the clients maintain copies of their root and swap files on a local computer mass storage device disk drive;

FIG. 2 is a simplified block diagram of a client-server computer system in accordance with the present invention wherein each client computer has an associated computer mass storage disk drive for use as a swap and local data cache for files accessed from the server computer over the network;

FIG. 3 is a simplified flowchart of a boot-up procedure for implementing the system and method of the present inven-

tion by building the file systems for a cache and swap space on a local disk drive of a client computer;

FIG. 4 is a simplified flowchart of a "read" operation directed to the data cache of a client computer utilizing the local disk drive as a cache in accordance with the present invention; and

FIG. 5 is a further, simplified flowchart of a "write" operation to the cache of a client computer utilizing the local disk drive as a data cache wherein the data is also initially written-through the cache to the network server computer.

DESCRIPTION OF A PREFERRED EMBODIMENT

The environment in which the present invention is used encompasses the general distributed computing system, wherein general purpose computers, workstations or personal computers are connected via communication links of various types, in a client-server arrangement, wherein programs and data, many in the form of objects, are made available by various members of the system for execution and access by other members of the system. Some of the elements of a general purpose workstation computer are shown in FIG. 1A, wherein a processor 1 is shown, having an input/output ("I/O") section 2, a central processing unit ("CPU") 3 and a memory section 4. The I/O section 2 is connected to a keyboard 5, a display unit 6, a disk storage unit 9 and a compact disk read only memory ("CDROM") drive unit 7. The CDROM unit 7 can read a CDROM medium 8 which typically contains programs 11 and data. The computer program products containing mechanisms to effectuate the apparatus and methods of the present invention may reside in the memory section 4, or on a disk storage unit 9 or on the CDROM 8 of such a system.

With reference now to FIG. 1B, a conventional "diskless" client-server configuration 10 is shown for purposes of comparison with the system and method of the present invention hereinafter described in more detail. The diskless client-server configuration 10 comprises, in pertinent part, a number of diskless client computers ("clients") 12 (denominated C_1 and C_2) bi-directionally interconnected for exchange of data with a server computer ("server") 14 by means of a network 16.

System-wide computer mass storage of data, files, application software and the like is maintained remotely from the diskless clients 12 and is associated directly with the server 14. In the simplified representative illustration shown, the client root file storage 18, /usr file storage 20, client data storage 22 and client swap area 24 is shown schematically as resident on various separate computer mass storage devices such as Winchester, "fixed", "rigid" or "hard" disk drives or related subsystems. The respective root, /usr, client data files and swap area may, of course, be physically stored together or separately on one or more computer mass storage device disk drives, subsystems or other storage media as is appropriate to the particular client-server system.

As is implied by its name, a diskless client 12 does not have an associated local disk drive and all client data resides on its server 14. The strongest aspect of the conventional diskless client-server configuration 10 is its relative ease of administration and diskless clients 12 are FRUs, do not require individual backup, their data can be modified from a central server 14 and software installation and upgrading is easily and quickly accomplished at the server 14.

On the other hand, the performance of the conventional diskless client-server configuration 10 is relatively poor and does not scale well. As additional diskless clients 12 are added to a network 16, the performance of all existing diskless clients 12 (as well as the server 14) is degraded. This performance degradation generally becomes intolerable before a reasonable number of clients 12 per server 14 may be reached. As to disk drive usage and cost, the conventional diskless client server configuration 10 has a meaningful advantage over other client-server configurations in that the individual diskless clients 12 do not have a local disk drive and are therefore less expensive to purchase than a comparable machine with an associated hard drive.

With reference additionally now to FIG. 1C, a conventional "dataless" client-server configuration 30 is also shown for purposes of comparison with the system and method of the present invention hereinafter described in more detail. The dataless client-server configuration 30 comprises, in pertinent part, a number of dataless client computers ("clients") 32 (again denominated C_1 and C_2) bi-directionally interconnected for exchange of data with a server computer ("server") 34 by means of a network 36.

As with the diskless system of FIG. 1B, system-wide computer mass storage of data, files, application software and the like is also maintained remotely from the dataless clients 32 and is again associated directly with the server 34. In the simplified representative illustration shown, the /usr file storage 40 and client data storage 42 is shown schematically as resident on separate computer mass storage devices or subsystems. As before, they may, of course, be physically stored together or separately on one or more computer mass storage device disk drives, subsystems or other storage media as is appropriate to the particular client-server system.

The conventional dataless client-server configuration 30 is distinguished from the diskless client-server configuration 10 of FIG. 1B by the addition of swap and local root file storage 44 associated with each of the dataless clients 32. The local root file storage 44 contains a copy of the local root file system from the root file storage 38. The /usr file system from the /usr file storage 40 is mounted from the central server 34 and may be shared with other dataless clients 32.

A dataless client-server configuration 30 is relatively easy to administer although not as easy as the diskless client-server configuration 10 shown in FIG. 1B. For example, a dataless client 32 is not an FRU, it must be individually backed up and installation is relatively more complex. Performance of a dataless client-server configuration 30 is generally adequate and is much more scaleable than diskless configurations. Nevertheless, dataless clients 32 will still tend to contend with one another for access to the /usr file system of the /usr file storage 40 associated with the server 34. Computer mass storage cost is relatively low in this configuration and a dataless client 32 can use a relatively small and inexpensive local disk drive of on the order of 100 MBytes capacity for many current applications.

With reference now to FIG. 2, a simplified block diagram of a system and method for implementing a cache only client-server computer configuration 50 utilizing a local client disk drive as a data cache in accordance with the present invention is shown. The cache only client-server configuration 50 comprises, in pertinent part, a number of cache only client computers ("clients") 52 (also again denominated C_1 and C_2) bi-directionally interconnected with a server computer ("server") 54 by means of a network 56. In the configuration illustrated, the network 56 may utilize, for example, the network file system software, NFS® available from Sun Microsystems, Inc., assignee of the present invention.

System-wide computer mass storage of data, files, application software and the like is again maintained remotely

from the cache only clients **52** and is associated directly with the server **54**. In the simplified representative illustration shown, the client root file storage **58**, /usr file storage **60** and client data storage **62** is again shown schematically as resident on separate computer mass storage devices or subsystems. As before, they may, of course, be physically stored separately or together on one or more computer mass storage device disk drives or other storage media as is appropriate to the particular client-server system.

The cache only client-server configuration **50** is distinguished from the diskless client-server configuration **10** of FIG. **1B** by the addition of a swap area and local data cache **64** associated with each of the cache only clients **52** and from the dataless client-server configuration **30** of FIG. **1C** by the fact that the /usr files are not the only files which might be stored locally at the client.

Cache only clients **52** may use relatively inexpensive local disk drives as small as 100 MBytes for swap and a local data cache **64** of recently used files from the root and /usr file systems accessed over the network **56**. The combination of local swapping and local disk caching of frequently used files leads to a dramatic reduction in overall network traffic as compared to the conventional diskless client-server configuration **10** of FIG. **1B**. Other read-mostly file systems, for example the /usr/local, could also be mounted in the same local data cache **64**. More specific information regarding setting up these mounts in a specific implementation of the cache only client-server configuration **50** utilizing the Solaris™ OS and the CacheFS™ file system available from Sun Microsystems, Inc., 2550 Garcia Avenue, Mountain View, Calif. 94043-1100, assignee of the present invention, is described in the: Cache File System (CacheFS) White Paper, Revision A, February 1994.

The server **54** may be configured much like a conventional diskless or dataless server as shown in the preceding figures. There is a small root area for each cache only client **52**, shown as root file system **58**, typically requiring on the order of 15–20 MBytes of disk space per cache only client **52**. The /usr file system maintained in the /usr files storage **60** is shared by all cache only clients **52** of the same architecture and OS revision. The server **54** may be configured to support cache only clients **52** of mixed architectures and/or mixed OS revisions. Unlike the conventional diskless client-server configuration **10** of FIG. **1B**, there is no client swap space on the server **54**. Since client swap files are generally significantly larger than client root directories, the cache only client-server configuration **50** achieves a substantial savings in server **54** disk space as compared to diskless systems and the server **54** need not be a particularly high powered computer. In a particular embodiment of a cache only client-server configuration **50** utilizing a SPARCstation™ IPC™ (trademarks of Sun Microsystems, Inc.) server **54** with 16 MBytes of random access memory ("RAM") was an effective server for more than thirty cache only client computers **52**.

The local data cache **64** used by a cache only client **52** is strictly a "write-through" cache and all modifications to data are reflected directly to the server **54**. Thus, there is never any critical data resident only at the cache only client **52**. This fact presents two major administrative advantages in that there is no need to individually back up (or archive) data from the cache only clients **52** and they are FRUs. Moreover, it is easy to replace a given cache only client **52** with another client computer in the event of hardware upgrade or hardware failure. The replacement cache only client **52** assumes the "identity" of the client it replaces and all local configuration and customization, such as printer configuration, locally defined passwords and the like is preserved. When the replacement cache only client **52** is booted, its local data cache **64** is populated with the data that the previous client had stored on the server **54**.

In a particular embodiment of the present invention implemented utilizing the Solaris OS, installation of the system may be effectuated utilizing the accompanying "install" technology. If the server **54** is to support cache only clients **52** of varying architectures, the architecture specific support may be selected at install time or installed after the fact using [swmtool]. The Solaris OS install program reserves disk space for the client root areas after asking how many clients will be supported. Like a diskless client **12** (FIG. **1B**), a typical cache only client root area will consume on the order of 15–20 MBytes of server **54** disk space.

In a particular implementation of the cache only client-server configuration **50**, the installation of a cache only client **52** may be performed entirely on the server **54** using the Solstice AutoClient™ manager application also available from Sun Microsystems, Inc. The user of the Solstice AutoClient manager, which is typically the system administrator, need only respond to a few simple questions per cache only client **52** relating to host name, IP address, ethernet address, disk configuration, swap size and the like. The Solstice AutoClient manager further supports the addition of cache only clients **52** in "batches" allowing the administrator to complete all interaction before any of the cache only clients **52** are actually installed rather than waiting for one cache only client **52** to complete before adding the next one. Moreover, the cache only client **52** hardware need not be available, or even powered on, during the installation process. After the installation process completes, the cache only client **52** simply does a "boot net".

A cache only client **52** implements a network boot. The early part of the boot (loading the kernel) is always done over the network **56**. Early in the boot process, the local data cache **64** is configured and "plugged in." At this point, population of a newly created local data cache **64** can begin. If this process involves a reboot of a machine that already had a previously created local data cache **64**, files in it may be used at this point. Using files from the local data cache **64** instead of retrieving them over the network **56** results in a dramatic reduction in network **56** traffic and server **54** load.

With reference additionally now to FIG. **3**, a representative process flow for enabling a cache only client during a boot sequence **70** is shown. The boot sequence **70** begins at start step **72** and then, at decision step **74**, a determination is made as to whether the client computer is intended to be a cache only client **52** or have an alternative configuration with respect to the server **54** such as a diskless client **12** (FIG. **1B**), dataless client **32** (FIG. **1C**) or stand-alone computer. If the client computer is not to be configured as a cache only client **54**, the process proceeds directly to end step **94** as shown.

Alternatively, if the client computer is to be configured as a cache only client **52**, then the boot sequence **70** continues to decision step **76** where a determination is made as to whether or not the client computer has been previously configured as a cache only client **52** due to the presence or absence of a previously implemented local data cache **64**. If a previous local data cache **64** is not found, the steps to implement a new cache are begun at step **78** and the attached computer mass storage disk drive is located at step **80**. Once the local disk drive has been identified, the file systems for the cache and swap are constructed at step **82**.

On the other hand, if at decision step **76**, a previous local data cache **64** is found, the steps to implement the previous

local data cache **84** is begun at step **84**. The previous local data cache **64** is tested at decision step **86** as to the integrity of the existing file system. If the integrity of the file system does not check out, the boot sequence **70** proceeds to step **82** to build the necessary file systems for the cache and swap as previously described, then to proceed to decision step **88**. Should the file integrity of the previous local data cache **64** be confirmed, the boot sequence proceeds directly to decision step **88** to test the integrity of the cache data structures on the local disk drive.

If the integrity of the cache data structures does not check out, the requisite cache data structures are created on the attached local disk drive at step **92** and the cache enabled at step **90**. Alternatively, if the integrity of the cache data structures is confirmed at decision step **88**, the boot sequence **70** continues directly to step **90** to enable the cache. In either event, the boot sequence **70** concludes at step **94**.

With reference additionally now to FIG. **4**, a representative "read" sequence **100** is shown as may be conducted by a cache only client **52**. The read sequence **100** begins at start step **102** and proceeds to a decision step **104** to determine whether or not the data requested to be read is already in the local data cache **64** (a cache "hit") or is not currently in the local data cache **64** (a cache "miss"). If the requested data is in the local data cache **64**, the read sequence **100** continues to decision step **106** to determine whether or not the data in the cache is consistent. If the read operation is a cache hit at decision step **104** and is determined to be consistent at decision step **106**, the read sequence **100** proceeds to step **108** to return the cached data requested and the read sequence **100** concludes at end step **114**.

On the other hand, if at decision step **104** the data is not currently in the local data cache **64**, or at decision step **106** the data is in the local data cache **64** but is not consistent, the read sequence continues at step **109**.

At step **109** the cache data for the particular file to be read is invalidated and space is allocated in the local data cache **64** for the requested data at step **110**. The data read from the server **54** over the network **56** is then copied to the local data cache **64** at step **112** and the cached data returned at step **108** to proceed to end step **114**.

With further reference additionally now to FIG. **5**, a representative "write" sequence **120** is shown. Write sequence **120** begins at start step **122** and proceeds to step **124** wherein the modified data to be written is first written directly to the server **54** over the network **56**. Following this "write-around" operation, the modified data is then written to the local data cache **64** at step **126** and the write sequence **120** concludes at end step **128**. As can be seen, in a write operation, the cache only client **52** may make no determination as to whether or not the modified data may already exist in the local data cache **64** and all "writes" are first to the server **54** and then to the local data cache without regard to a cache "hit" or "miss". In this manner, the modified data is always available at the server **54** for ease of system administration.

In operation, files in the root and /usr file systems from the root file storage **58** and /usr file storage **60** respectively, are copied to the local data cache **64** disk drive as they are referenced. Virtually any action on a file will cause it to be copied to the local data cache **64** such as invocation of an executable file, read of a text file and the like. In this regard, it's important to note that large portions of the OS distribution are never accessed by most users and this means that this data is never copied to their local data cache **64** disk drive. Thus, the local data cache **64** disk drive can be much smaller in capacity than the disk drive that would be required for a local installation of the OS.

Once data has been copied to the local data cache **64**, no server **54** interaction is required to reference it again other than as noted hereinafter. A subsequent access to cached data may be faster than a network **56** access, and results in a reduction in the load placed on the server **54** and the network **56**.

With the Solstice AutoClient implementation of a cache only client **52**, a new cache consistency mode has been added to the CacheFS consistency model described in the aforedescribed CacheFS white paper. This consistency mode is called [demandconst]. This mode assumes that files are not generally being changed on the server **54**, and that if they ever are changed someone (typically the system administrator) will explicitly request a consistency (or cache coherency) check. Thus, no consistency checking is performed unless a check is explicitly requested. Nevertheless, there is an implied consistency check when a CacheFS file system is mounted (when the Solstice AutoClient client boots) and a Solstice AutoClient cache only client **52** is configured by default to request a consistency check every 24 hours via [cron(1)]. This model improves Solstice AutoClient cache only client **52** performance as a lot of network **56** latency is avoided by skipping consistency checking. The risk of inconsistent data is minimal as the cache only client **52**'s root area is exported only to that particular cache only client **52**. There is no local data cache **64** inconsistency when the cache only client **52** modifies its own data since such modifications are made through the local data cache **64**. The only other way a cache only client **52**'s root data can be modified is by the super user on the server **54**. Presumably this operation would be done by the system administrator when, for example, installing new software. The /usr file system in the /usr file storage **60** is similar in that the server **54** exports it read-only, so the only way it is likely to be modified is by the system administrator on the server **54**. In these cases, it is reasonable to require the system administrator to initiate a consistency check on the cache only client **52**'s behalf. The [autosync(1 m)] command is provided for this purpose and may be run on the cache only clients **52** as well as the server **54** in case the cache only clients **52** cannot wait for the system administrator's intervention or the 24 hour interval.

Except for enhanced performance, a cache only client **52** functions no differently from an end user's viewpoint than any other client-server configuration and a cache only client **52** appears to be just another computer with all conventional OS capabilities being supported. In a particular embodiment of the present invention utilizing the Solstice AutoClient product, a cache only client **52** was booted, openwindows started, and xterm, mailtool, calendar manager, xclock and FrameMaker™ were run while filling only 24 MBytes of local data cache **64** space.

In benchmark testing running desktop applications such as mailtool, cmdtool, calendar manager, file manager, xterm and xclock, a stand-alone configuration completed the benchmark in the least time. However, the average time for a cache only client-server configuration **50** was only 1.27 times that of the stand-alone configuration. On the other hand, the average time for a conventional dataless client-server configuration **30** (FIG. 1C) was 1.5 times that of the stand-alone configuration while the average time for a conventional diskless client-server configuration **10** (FIG. 1B) was 4 times that of the same stand-alone setup.

In general, the cache only client-server configuration **50** provides great ease of system administration and since the

cache only clients 52 are FRUs, they don't require individual backup, their data can be modified from a central server 54, and installation is easily and quickly accomplished from the server 54. As noted above, the overall system performance achieved is very good and the use of a local disk drive for swap and caching of frequently accessed files provides a substantial performance boost over diskless systems. Performance is also better than dataless systems since the local data cache 64 reduces server 54 requests for /usr files. The cache only client 52 can use a relatively small capacity local disk drive of on the order of 100 MBytes for the local data cache 64.

While there have been described above the principles of the present invention in conjunction with specific computer hardware, operating and network file systems software and overall system configurations and process flows, it is to be clearly understood that the foregoing description is made only by way of example and not as a limitation to the scope of the invention.

What is claimed is:

1. A client-server computer system comprising:

at least one server computer coupled to a network for transferring and receiving data thereover, said server computer having at least one associated server mass storage device to which said data may be written and from which said data may be read; and

at least one client computer coupled to said network for transferring and receiving said data thereover, said client computer having at least one associated client mass storage device to which said data may be written and from which said data may be read;

root files associated with said client computer and stored in said server mass storage device, wherein said client mass storage device is at least partially configurable as a local data cache for at least some of said root files.

2. The client-server computer system of claim 1 wherein a requested data access initiated by said client computer will be directed to said local data cache.

3. The client-server computer system of claim 2 wherein a read access operation of said requested data will be directed to said local data cache if said requested data is currently in said local data cache.

4. The client-server computer system of claim 2 wherein a read operation of said requested data will be directed to said server if said requested data is not currently in said local data cache.

5. The client-server computer system of claim 2 wherein a read operation of said requested data will be directed to said server if said requested data is currently in said local data cache but is not coherent with said requested data at said server computer.

6. The client-server computer system of claim 4 wherein said requested data obtained from said server is further written to said local data cache.

7. The client-server computer system of claim 2 wherein a write access operation of modified data will be directed to said server computer and said modified data also written to said local data cache.

8. The client-server computer system of claim 6 wherein a subsequent read access operation of said modified data will be directed to said local data cache.

9. The client-server computer system of claim 1 wherein said client computer is capable of initiating a network boot operation in response to which said server computer transfers at least a portion of an operating system over said network to said local data cache in said client mass storage device to effectuate said boot of said client computer whereupon data subsequently read from said server mass storage device over said network by said client computer is also written to said data cache.

10. The client-server computer system of claim 1 wherein modified data written to said server mass storage device over said network by said client computer is also written to said data cache.

11. A process for accessing data in a client-server computer system having at least one server computer and at least one client computer coupled to a network for transferring and receiving information thereover wherein said server computer and said client computer each have at least one associated server and client mass storage device respectively to which data may be written and from which data may be read, said client mass storage device being at least partially configurable as a local data cache, the process comprising the steps of:

providing root files associated with said client computer and stored in said server mass storage device;

providing for initiation of a requested data read operation by said client computer to access said root files;

providing for determining whether said requested data is currently in said local data cache; and

providing for returning said requested data to said client computer from said local data cache if said requested data is currently in said local data cache.

12. The process of claim 11 further comprising the step of:

providing for requesting said requested data from said server computer if said requested data is not currently in said data cache.

13. The process of claim 12 further comprising the step of:

providing for writing said requested data from said server computer to said local data cache following said step of providing for requesting.

14. The process of claim 13 further comprising the step of:

providing for allocating space in said local data cache for said requested data prior to said step of providing for writing.

15. The process of claim 11 further comprising the step of:

providing for determining a coherency between said requested data and said requested data in said local data cache prior to said step of providing for returning.

16. A process for establishing a data cache associated with a client computer in a client-server computer system, said client computer having at least one client mass storage device to which data may be written and from which data may be read, said client mass storage device being at least partially configurable as said data cache, said process comprising the steps of:

providing for locating of said client mass storage device;

providing for determining a presence of a previously created data cache;

providing for building a file system for said data cache; and

providing for enabling of said data cache.

17. The process of claim 16 further comprising the steps of:

providing for determining a presence of a previously created data cache;

providing for checking an integrity of one or more file systems of said previously created data cache; and

providing for utilization of said previously created data cache.

18. The process of claim 16 further comprising the step of:

providing for checking an integrity of one or more cache data structures on said client mass storage device; and

providing for creating of said one or more cache data structures prior to said step of providing for enabling of said data cache.

19. A client computer couplable to a server computer by means of a network for intercommunication of data therebetween, said client computer capable of performing a network boot operation wherein at least a portion of an operating system is transferred to said client computer from said server computer over said network, said client computer comprising:

a local mass storage device capable of being at least partially configured as a data cache;

wherein said network boot operation initiated by said client computer effectuates transfer of said portion of said operating system from said server to said client computer and copying of said portion of said operating system to said data cache such that subsequent requests for said portion of said operating system are directed to said data cache.

20. The client computer of claim 19 wherein a data write operation initiated by said client computer effectuates transfer of modified data to said server and copying of said modified data to said data cache.

21. A computer program product comprising:

a computer usable medium having computer readable code embodied therein for implementing accessing of data in a client-server computer system having at least one server computer and at least one client computer coupled to a network, said server and client computers each having at least one associated server and client mass storage device respectively to which data may be written and from which data may be read, said client mass storage device being at least partially configurable as a local data cache;

computer readable program code devices configured to cause a computer to provide root files associated with said client computer and stored in said server mass storage device;

computer readable program code devices configured to cause a computer to effect initiation of a requested data read operation by said client computer to access said root files;

computer readable program code devices configured to cause a computer to effect determination of whether said requested data is currently in said local data cache; and

computer readable program code devices configured to cause a computer to effect returning of said requested data to said client computer from said local data cache if said requested data is currently in said local data cache.

22. The computer readable program code devices of claim 21 further comprising:

computer readable program code devices configured to cause a computer to effect requesting of data from said server computer if said requested data is not currently in said data cache.

23. The computer readable program code devices of claim 21 further comprising:

computer readable program code devices configured to cause a computer to effect writing said requested data from said server computer to said local data cache.

24. The computer readable program code devices of claim 22 further comprising:

computer readable program code devices configured to cause a computer to effect allocation of space in said local data cache for said requested data.

25. The computer readable program code devices of claim 21 further comprising:

computer readable program code devices configured to cause a computer to effect determination of coherency between said requested data and said requested data in said local data cache.

26. A computer program product comprising:

a computer usable medium having computer readable code embodied therein for implementing an establishment of a data cache associated with a client computer in a client-server computer system, said client computer having at least one client mass storage device to which data may be written and from which data may be read, said client mass storage device being at least partially configurable as said data cache;

computer readable program code devices configured to cause a computer to effect location of said client mass storage device;

computer readable program code devices configured to cause a computer to effect determination of a presence of a previously created data cache;

computer readable program code devices configured to cause a computer to effect building of a file system for said data cache; and

computer readable program code devices configured to cause a computer to effect enablement of said data cache.

27. The computer program product of claim 26 further comprising:

computer readable program code devices configured to cause a computer to effect checking an integrity of one or more file systems of said previously created data cache; and

computer readable program code devices configured to cause a computer to effect utilization of said previously created data cache.

28. The computer program product of claim 27 further comprising:

computer readable program code devices configured to cause a computer to effect checking an integrity of one or more cache data structures on said client mass storage device; and

computer readable program code devices configured to cause a computer to effect creation of said one or more cache data structures.

* * * * *

US005764906A

United States Patent [19]

Edelstein et al.

[11] Patent Number: 5,764,906

[45] Date of Patent: Jun. 9, 1998

[54] **UNIVERSAL ELECTRONIC RESOURCE DENOTATION, REQUEST AND DELIVERY SYSTEM**

[75] Inventors: **Matthew Edelstein**, Arlington; **Samuel Bergman**, Alexandria, both of Va.; **Donald B. Rubin**, Silver Spring, Md.

[73] Assignee: **Networld LLC**, Arlington, Va.

[21] Appl. No.: **554,775**

[22] Filed: **Nov. 7, 1995**

[51] **Int. Cl.**[6] **G06F 13/14**

[52] **U.S. Cl.** **395/200.49**; 395/200.55; 395/200.33

[58] **Field of Search** 395/200.01, 200.02, 395/200.1, 200.09, 200.49, 200.55, 200.33, 200.8, 200.58

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,677,588	6/1987	Benjamin et al.	395/200.58
4,718,005	1/1988	Feigenbaum et al.	395/200.1
5,282,244	1/1994	Fuller et al.	379/230
5,388,213	2/1995	Oppenheimer et al.	395/200.01
5,434,974	7/1995	Loucks et al.	395/612
5,586,257	12/1996	Perlman	463/42

OTHER PUBLICATIONS

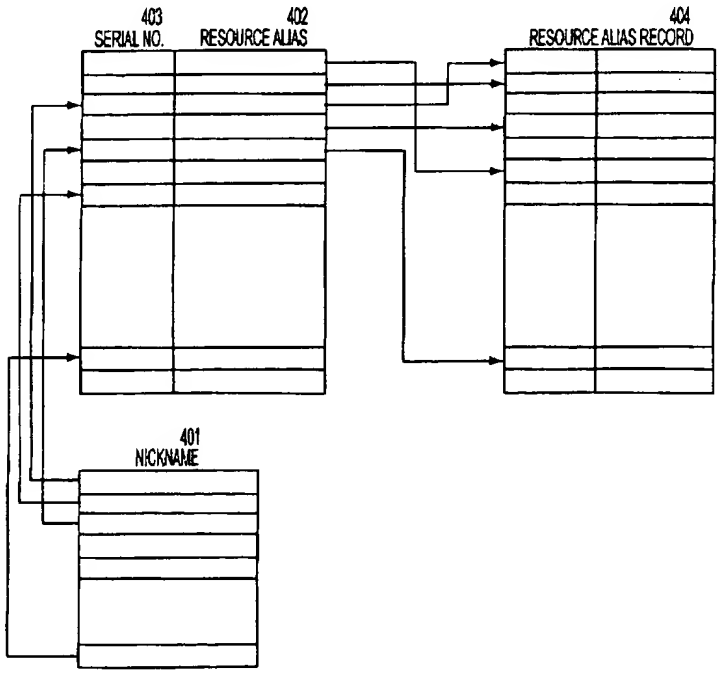
A practical method for achieving portable communications in the internet context by Qu et al., 1995 IEEE publication, pp. 1512–1516.

Primary Examiner—Parshotam S. Lall
Assistant Examiner—Z. Maung
Attorney, Agent, or Firm—Fulbright & Jaworski LLP

[57] **ABSTRACT**

A universal electronic resource denotation, request and delivery system allows a user to locate information on a distributed computer system or network such as the Internet by knowing or guessing a short mnemonic alias of an electronic resource without the user having to know the physical or other location denotation such as the universal resource locator (URL) of the desired resource. The system hardware includes a client computer, a local server, a central registry server, a value added server, and a root server. The universal electronic resource denotation, request and delivery system supports a personal aliasing (nicknaming) feature, a universal resource accessing feature for finding location information such as URLs relating to a query term, a "see also" feature for including information about related documents or resources within the record of a resource, a feature for updating local servers and client machines by periodically deleting those records which have changed, a "try again" and "mirroring" feature for aiding a user in obtaining the resource under adverse hardware or software conditions, and an authentication and administration feature that allows a user to administer the aliases and related data which pertain to his/her resources.

31 Claims, 6 Drawing Sheets



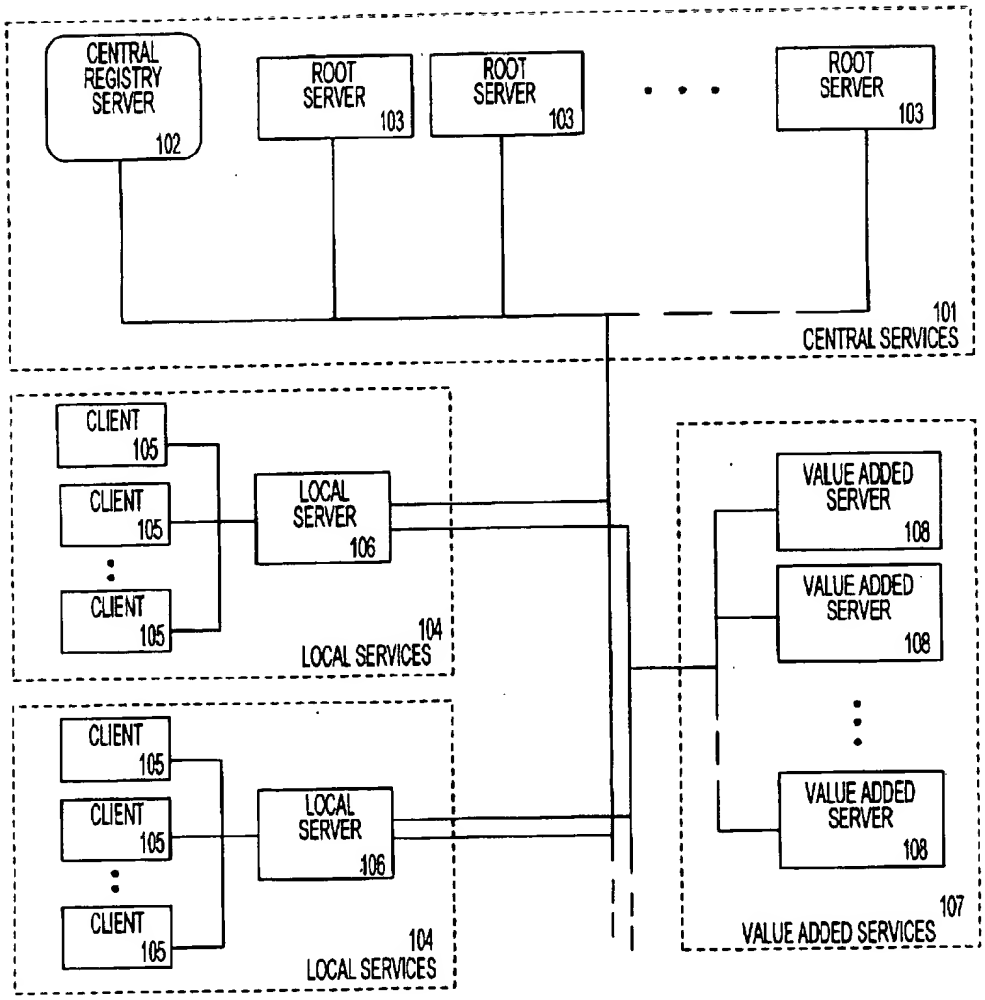

CENTRAL REGISTRY SERVER
102
ROOT SERVER
103
ROOT SERVER
103
ROOT SERVER
103
101
CENTRAL SERVICES
CLIENT
105
CLIENT
105
CLIENT
105
LOCAL SERVER
106
104
LOCAL SERVICES
CLIENT
105
CLIENT
105
CLIENT
105
LOCAL SERVER
106
104
LOCAL SERVICES
VALUE ADDED SERVER
108
VALUE ADDED SERVER
108
VALUE ADDED SERVER
108
107
VALUE ADDED SERVICES

FIG. 1

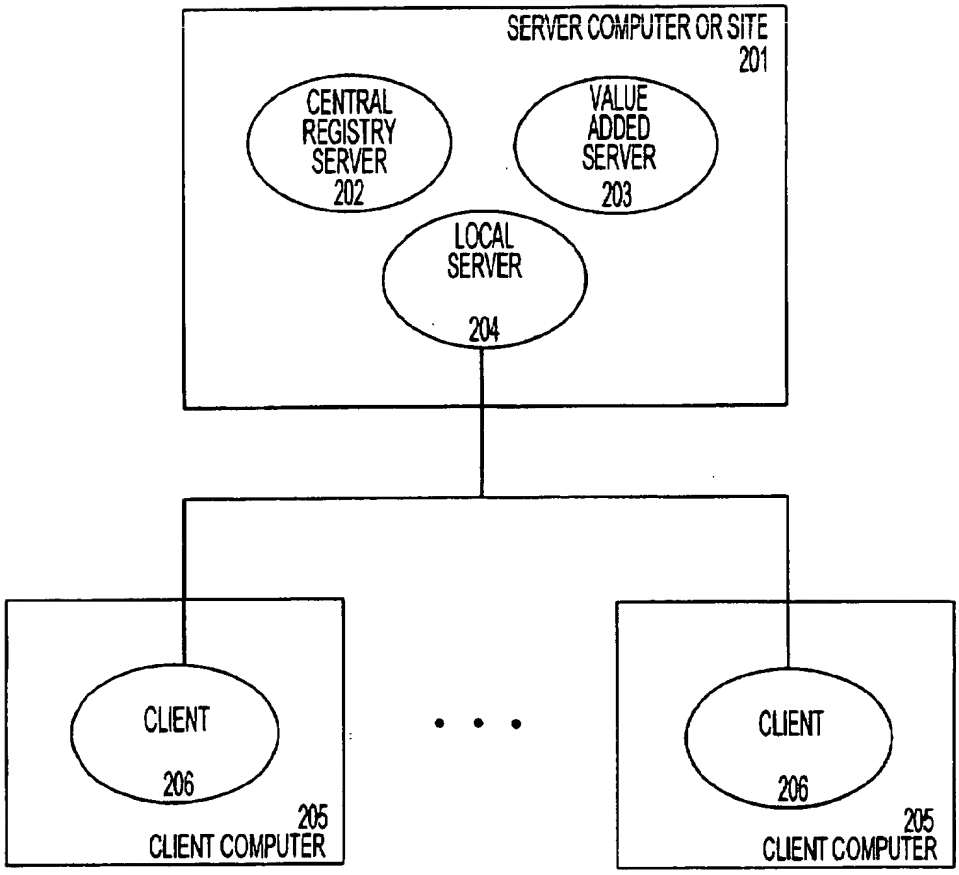

SERVER COMPUTER OR SITE
201
CENTRAL REGISTRY SERVER
202
VALUE ADDED SERVER
203
LOCAL SERVER
204
CLIENT
206
205
CLIENT COMPUTER
CLIENT
206
205
CLIENT COMPUTER

FIG. 2

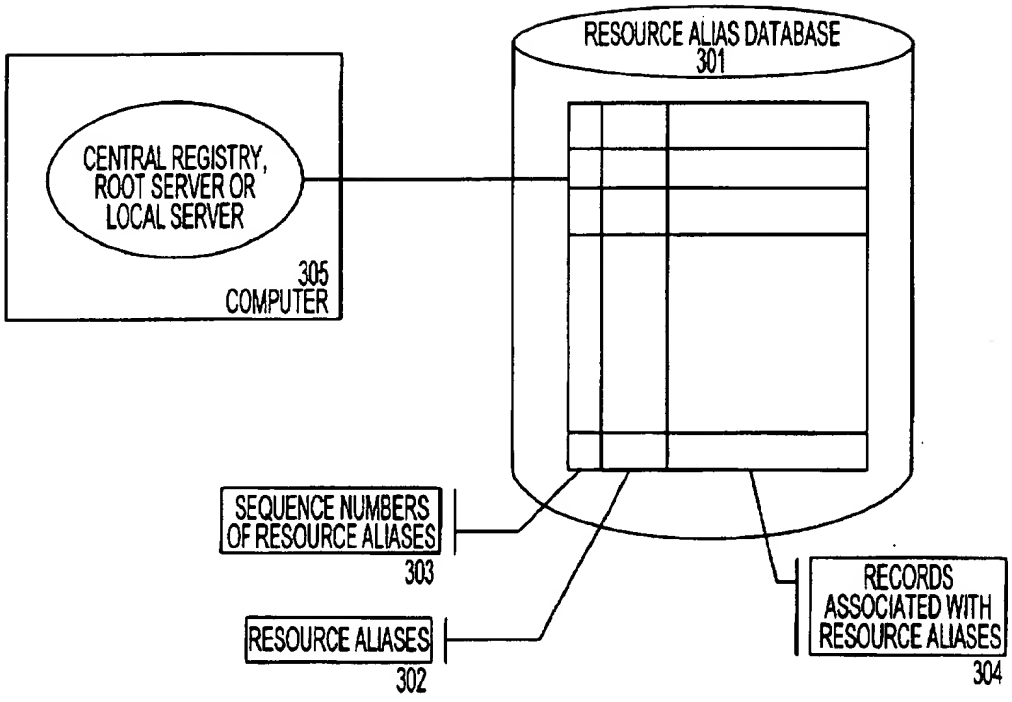

RESOURCE ALIAS DATABASE
301
CENTRAL REGISTRY,
ROOT SERVER OR
LOCAL SERVER
305
COMPUTER
SEQUENCE NUMBERS
OF RESOURCE ALIASES
303
RESOURCE ALIASES
302
RECORDS
ASSOCIATED WITH
RESOURCE ALIASES
304

FIG. 3

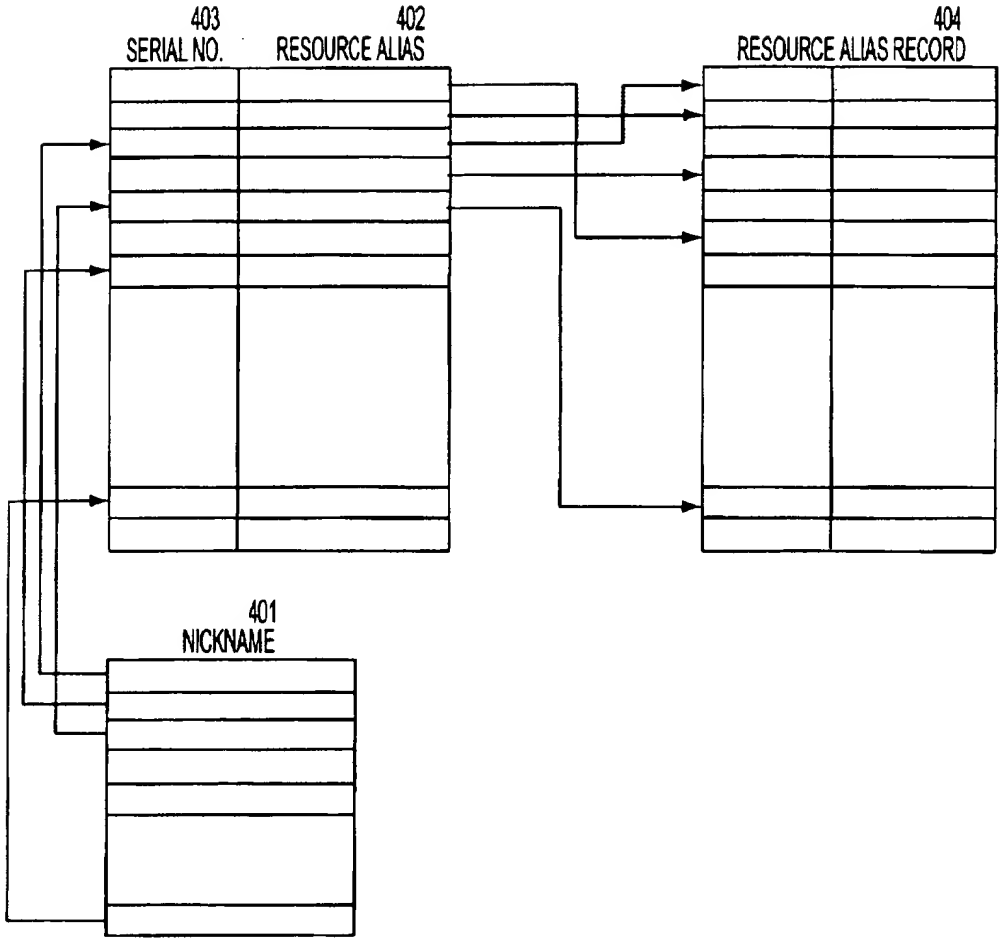

403
SERIAL NO.
402
RESOURCE ALIAS
404
RESOURCE ALIAS RECORD
401
NICKNAME

FIG. 4

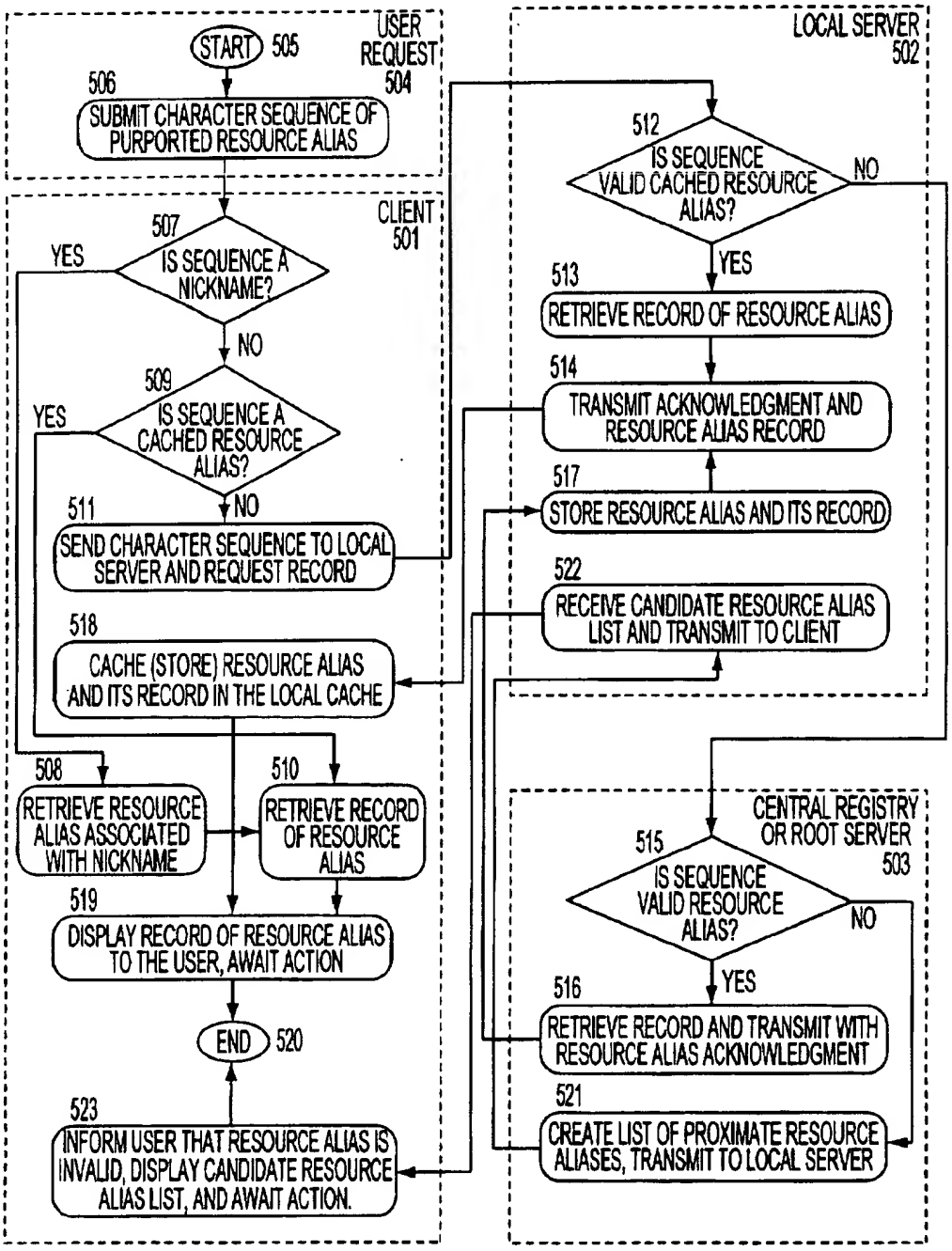

USER REQUEST 504
START 505
506
SUBMIT CHARACTER SEQUENCE OF PURPORTED RESOURCE ALIAS
CLIENT 501
507
IS SEQUENCE A NICKNAME?
YES
NO
509
IS SEQUENCE A CACHED RESOURCE ALIAS?
YES
NO
511
SEND CHARACTER SEQUENCE TO LOCAL SERVER AND REQUEST RECORD
518
CACHE (STORE) RESOURCE ALIAS AND ITS RECORD IN THE LOCAL CACHE
508
RETRIEVE RESOURCE ALIAS ASSOCIATED WITH NICKNAME
510
RETRIEVE RECORD OF RESOURCE ALIAS
519
DISPLAY RECORD OF RESOURCE ALIAS TO THE USER, AWAIT ACTION
END 520
523
INFORM USER THAT RESOURCE ALIAS IS INVALID, DISPLAY CANDIDATE RESOURCE ALIAS LIST, AND AWAIT ACTION.
LOCAL SERVER 502
512
IS SEQUENCE VALID CACHED RESOURCE ALIAS?
NO
YES
513
RETRIEVE RECORD OF RESOURCE ALIAS
514
TRANSMIT ACKNOWLEDGMENT AND RESOURCE ALIAS RECORD
517
STORE RESOURCE ALIAS AND ITS RECORD
522
RECEIVE CANDIDATE RESOURCE ALIAS LIST AND TRANSMIT TO CLIENT
CENTRAL REGISTRY OR ROOT SERVER 503
515
IS SEQUENCE VALID RESOURCE ALIAS?
NO
YES
516
RETRIEVE RECORD AND TRANSMIT WITH RESOURCE ALIAS ACKNOWLEDGMENT
521
CREATE LIST OF PROXIMATE RESOURCE ALIASES, TRANSMIT TO LOCAL SERVER

FIG. 5

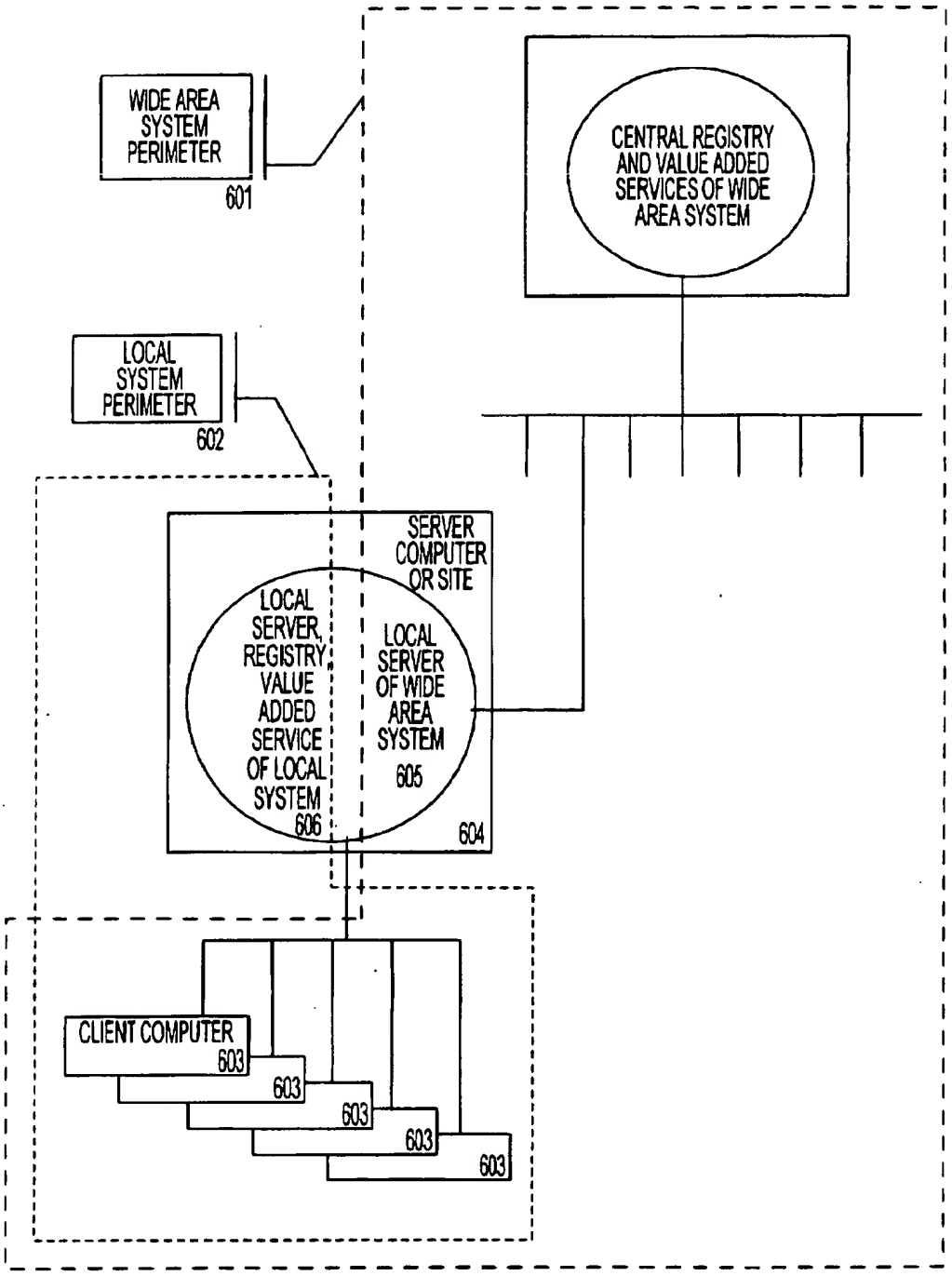

WIDE AREA SYSTEM PERIMETER
601
CENTRAL REGISTRY AND VALUE ADDED SERVICES OF WIDE AREA SYSTEM
LOCAL SYSTEM PERIMETER
602
SERVER COMPUTER OR SITE
LOCAL SERVER, REGISTRY VALUE ADDED SERVICE OF LOCAL SYSTEM
606
LOCAL SERVER OF WIDE AREA SYSTEM
605
604
CLIENT COMPUTER
603
603
603
603
603

FIG. 6

UNIVERSAL ELECTRONIC RESOURCE DENOTATION, REQUEST AND DELIVERY SYSTEM

BACKGROUND OF THE INVENTION

1. Field of the Invention

The field of the invention is that of a Network in which Electronic Resources are shared among such a system's community of users. A Network is a distributed communicating system of computers which are interconnected by various electronic communication links and computer software protocols. Electronic Resources are (a) documents, files and/or other information sources which are accessible on a Network to its user community, or (b) references to and/or means of accessing such documents, files and/or information sources. The invention relates to a system for denoting (naming) the Electronic Resources of a Network and a related system for the fulfillment of requests for and/or execution of delivery of these Electronic Resources to users of the Network's user community.

2. Description of the Related Technology

A particularly well-known Network is the international information infrastructure, commonly called the Internet. The Internet is a world-wide Network whose Electronic Resources include (but are not limited to) text files, graphic files in various formats, World Wide Web "pages" in HTML (HyperText Mark-Up Language) format, files in various and arbitrary binary formats, and electronic mail addresses. As in many other Networks, the scheme for denotation of an Electronic Resource on the Internet is an "electronic address" which uniquely identifies its location within the network and within the computer in which it resides. On the Internet, for example, such an electronic address is called a Universal Resource Locator or URL, and consists of a specially formatted concatenation of information about the type of protocol needed to access the resource, a Network Domain identifier, identification of the particular computer on which the Electronic Resource is located, a port number, directory path information within the computer's file structure, and the file name of the resource.

Internet URLs and similar denotation schemes for Electronic Resources are cumbersome for human users. URLs are often more than 50 characters long and contain information which is neither interesting nor meaningful to seekers of information. For example, the NASA Internet web "homepage" has the URL "http://hypatia.gsfe.nasa.gov/NASA_homepage.html."

The National Information Control (NIC) registers unique Internet domain names on a first come first served basis. Even if an entity can acquire a domain name which is mnemonic and easy to remember, the URLs associated with that domain may still be complex non-intuitive character sequences. Sometimes an entity cannot register a desired domain name because another entity has already registered that domain name with the NIC. The entity must then choose an alternate domain name, which often results in difficulty in finding that entity's Electronic Resources on the Internet. Additionally, due to length, software, and practicality considerations, domain names are often peculiar abbreviations, presenting additional confusion in locating an entity's Electronic Resources on the Internet. Furthermore, many entities do not possess their own computing equipment nor domain names, and must maintain their Electronic Resources in the domains of other entities, as in this example of a URL: "http://draco.centerline.com:8080/~franl/crypto.html," which denotes a web page on privacy and cryptography.

Thus, several difficulties face users attempting to locate Electronic Resources on a Network with a denotation system such as that in use on the Internet. They include the length, complexity and non-intuitive nature of denotations (URLs on the Internet), the need to type them correctly, and the difficulty in learning of the existence of the Electronic Resource and in discovering its correct denotation or URL.

Various software tools to facilitate the search for URLs have been proposed or developed for use on the Internet. These include "Yellow Pages," "White Pages," and "Web Crawlers." They all deal with compiling and maintaining classification systems of Electronic Resources on a Network. They all attempt to create and/or maintain a utility which presents an indexing scheme to a user so that he/she may learn of the existence of an Electronic Resource and retrieve its electronic address (or URL). "Yellow Page" indexes classify Electronic Resources by a hierarchy of subject areas in a manner similar to the telephone "yellow pages" or a library classification scheme. "White Page" indexes classify Electronic Resources by owner or name of the resource. These schemes inherit all the difficulties of classifying potentially huge name spaces, including the difficulties arising from overlapping and non-hierarchical subject areas and overlapping name spaces.

The Yellow Page approach suffers from the phenomenon of overlapping subject areas which occurs in any classification scheme. This can be illustrated by the difficulty of deciding whether to place "educational psychology" under "education" or "psychology" and that of classifying a document on Democracy and Fascism in Spain under possibly disparate subject areas of "democracy" or "fascism" or "Spanish history" in a classification scheme. These difficulties are well known in Library Science. The White Page approach is to classify by provider or owner. This is an excellent scheme providing that the information seeker knows the name(s) of potential providers of Electronic Resources. Neither of these schemes address the problem of complexity of the denotation of Electronic Resources. In some cases, the denotation need not be seen or dealt with by the user, as in the case of hypertext links ("hot links") within Internet Web pages. Web software automatically retrieves documents referred to in other documents without user intervention or entry of URLs. Web software ("browsers") also are able to retain URLs of Web pages in a user-created (usually hierarchical) classification scheme, and list these by page titles. This capability allows users to revisit (retrieve at a later time) web pages previously retrieved, as they may have changed.

Web crawlers and other search engines attempt to create indexes of the yellow or white page variety, together with their attendant classification scheme, by continually traversing Electronic Resources in a Network and compiling information about the resources encountered. In an environment similar to the World Wide Web in which documents link to other documents, search engines are able to extract the links from documents in order to extend their search to other documents. Various means are used to extrapolate subject areas and other classification schemes, ranging from author-provided keyword or indexing information through expert system techniques for ferreting index information from textual context. These engines participate in the construction apparatus for indexes such as yellow or white pages.

Some Networks include protocols such as the Internet "Finger" for finding an electronic mail address of a person. These protocols and their attendant software have the drawback of being unable to search a very large e-mail address space, and thus require additional information for their

search, such as a Domain Name in the Internet. Other Internet protocols such as "Who Is" request registered information about different Domains (from NIC in the Internet).

Thus, there are many tools in Networks for locating and classifying Electronic Resources. All deal with using user-provided information regarding the subject matter, owner or electronic location of an Electronic Resource in order to identify its electronic address (URL in the Internet). Other tools attempt to create, update or extend such classification systems automatically by continually searching the Network's Electronic Resource space. Still other tools construct and retain user-classified lists of these addresses for later use.

SUMMARY OF THE INVENTION

Using current technology, an user of a resource-sharing distributed or networked computer system such as the Internet cannot quickly and conveniently locate and access a specific network resource unless the user knows the precise network address of the desired resource. Indexes assist users in locating such resources, but as information grows exponentially, indexes are larger and thus more difficult to traverse and may not be able to remain current. Information providers may choose to advertise electronic addresses of their resources (such as URLs on the Internet) by other means (newspapers, radio, television and other media, for example); electronic addresses advertised by these means are often cumbersome and difficult to remember, especially since they are often expressions of physical locations of resources and include non-intuitive arbitrary-seeming information. Seekers of information thrive on short mnemonic denotations for informational resources. In the telephone system this is demonstrated by competition for use of scarce alphabetically meaningful telephone numbers such as "800-FLOWERS," and on the Internet it is demonstrated by competition for mnemonic Domain names such as "www.ibm.com."

This invention deals with a mnemonic denotation system for Electronic Resources on a Network such as the Internet and a concomitant system of request and delivery services for these Electronic Resources. Specifically, this invention is a system for providing and maintaining short aliases for information resources and their providers and a system for translation of these aliases to meaningful electronic addresses such as URL's, facsimile and voice telephone numbers and electronic mail addresses, and for accessing the resources by means of these addresses.

The system according to the invention need not implement an information utility, nor need it store the information which information providers make available to user communities. Nor does a system according to the invention need to classify or index information in such a utility or network. This invention concerns itself primarily with a system for "aliasing" information resources with short mnemonic names chosen by the information providers and with a system for providing users with pointers to (access to) the information or actually delivering the information in a variety of formats on behalf of the information providers.

1. Definitions

A Network is a distributed communicating system of computers which are interconnected by various electronic communication links and computer software protocols for the purpose of sharing files, documents and other electronic resources among its community of users (such as the Internet).

An Electronic Resource is (a) a document, file and/or other information source which is accessible on a Network to its user community, or (b) a references to and implied means of accessing such a document, file and/or information source; thus, a text file, a web page, a Telnet connection, and an e-mail address are Electronic Resources; a URL on the Internet is also an Electronic Resource.

An Address is a character sequence which can be used directly to locate, communicate with and access an Electronic Resource, such as a URL on the Internet, an e-mail address, or a voice or facsimile telephone number.

A Denotation of an Electronic Resource is a name for that resource chosen by a naming convention delineated below.

A Request and Delivery System is the distributed software system delineated in this document for uncovering the "address" on an Electronic Resource in a Network, and for transporting or transmitting that resource in some desirable form to the requester.

Mnemonic means a denotation which is intended to be easy to remember for human users.

2. Objects of the Invention

An object of this invention is to a provide shortened, convenient, mnemonic method for denoting and accessing Electronic Resources on a Network such the Internet. Another object of this invention is to provide a distributed computer system that implements this method by associating (mapping) mnemonic denotations of Electronic Resources with their electronic addresses (such as URLs) and retrieving Addresses associated with the Denotations of the invention. Another object of this invention is to provide a mechanism for assuring that every Denotation of an Electronic Resource of a Network is unique within the Network and controlled by the owner and/or provider of the resource. Another object of this invention is to facilitate and/or provide a mechanism for the delivery of Electronic Resources associated with Denotations to users by electronic or other means. Still another object of this invention is to provide mechanism for users to identify the Denotations of Electronic Resources of an information provider. A further object of the invention is to provide a mechanism and distributed software system for guaranteeing that the mnemonic Denotations of Electronic Resources remain current and are continually associated with their correct physical or electronic Addresses and/or locations even when these Addresses or locations change.

3. Denotation Method for Information Resources

At the core of this invention is a system for associating any Electronic Resource within a Network with a special Denotation made up of a unique sequence of characters. For purposes of description herein, the Denotation of a particular Electronic Resource shall be referred to as its "Resource Alias." A Resource Alias includes a sequence of characters which together attempt to describe the resource in a mnemonically meaningful way. The system according to the invention may require a Resource Alias to always begin with a character sequence identifying the provider of an Electronic Resource which shall be referred to herein as a "Source Alias."

Source Aliases are made unique by constricting each Source Alias to exactly one information provider, be it an organization or an individual. A Source Alias uniquely describes and denotes the owner, source or distributer of the resource. Fictional examples of Source Aliases may be "US Senate," "IBM." "Harvard University" and "Red Cross." An information provider may create Resource Aliases by appending a separator character and another sequence of characters to his/her Source Alias. Each Resource Alias

uniquely denotes and represents an Electronic Resource, while many Electronic Resources may be associated with each Source Alias. Fictional examples of Resource Aliases might be "US Senate/Dole Bio" and "XYZ Appliance Company/four-slice toaster," wherein the separator character is 'slash' ('/'). The first example might be the Resource Alias associated with a biographical sketch of Senator Dole provided by the U.S. Senate, while the second might be a description of a particular toaster vended by the XYZ Appliance Company.

Source Aliases may be made unique by their registration in a central registry. Resource Aliases may be constructed by information providers. Both Source and Resource Aliases are expected to be Mnemonic and easy to convey to information seekers. It is also expected that entities will want to use short Source and Resource Aliases whenever possible. For example, the U.S. Department of Defense may desire to use "DoD" or "US DoD" as a Source Alias rather than (or in addition to) using its complete name. Some commonly used character strings may be reserved for use as Source Aliases or Resource Aliases by public entities and advertised as such; an example might be "911/poison."

In order to extend the name range for Source and Resource aliases, various non-alphabetic characters shall be allowed in their expression. Thus, for example, "Smith&Jones," "USA Today," and "CBS!" may be valid Source Aliases, while "MicroSoft/C++," "Boeing/767 Info." "IBM/PC=Value," and "Chase Manhattan/$Exchange" might be valid Resource Aliases. A system according to the invention may display a predetermined grammar and include a reserved vocabulary.

4. Central Registry of Denotations

According to the invention, there will be a central registry system for registration of Source Aliases. In addition, Resource Aliases may also be specified on a central registry system. Source Aliases would preferably be chosen on a first-come first-served basis, but there could be some criteria which prevents information providers from assuming Source Aliases which commonly indicate any well known public entity such as a national or international corporation or a non-profit or governmental agency. Aliases which denote information resources must be centrally registered for two reasons. First, in order to guarantee uniqueness, particularly of Source Aliases. Secondly, in order to allow Resource Aliases to be globally associated with the correct Addresses of the Electronic Resources they denote, and so that these associations may be made accessible to all users of the Network.

5. The Software and The Software Sites

The implementation of this invention comprises a distributed computer system including different computer and software implemented systems which communicate with each other on their Network. The description of this invention shall refer to various types of Network sites (computers) which are partially configured by and shall contain and operate according to different software programs. These sites shall be referred to as a "Central Registry" (the unique site which is responsible for registration of Source and Resource Aliases), "Clients" (computers such as workstations and personal computers used by human users), "Local Servers" (intermediate Network nodes to which Clients may be connected to and which provide immediate service for Clients), "Root Servers" (computers which contain essentially the same information as the Central Registry and provide this information to other computers so as to distribute the load which would otherwise fall on the Central Registry computer), and "Value Added Servers" (computers which provide additional services to Clients and their users as described below). The aggregate of software of this invention shall have component programs at each of these types of sites, and these component programs shall work in concert in order to provide the stated operational features to users and meet the stated objects of this invention.

The Central Registry is the site wherein the official versions of all Source and Resource Aliases, together with any relevant information associated with them, shall reside. It is the responsibility of this site to store and disseminate all Resource Alias-related information, to register new Source Aliases and Resource Aliases for information providers, and to allow Resource Alias-related information to be updated by information providers.

The Root Servers may be sites which "mirror" the Central Registry, in that they disseminate Resource Alias-related information. They are periodically updated by the Central Registry, but do not in themselves register Source or Resource Aliases, nor do they allow direct updating of Resource Alias-related data by information providers.

The Local Servers are so-called "nodes" in the Network in question. This type of site serves as an intermediary dissemination point for Resource Alias-related information. This type of site shall "cache" Resource Alias-related information (store such information for some period of time or while this information continues to be accessed at a reasonable rate). Thus, the aggregate of Resource Alias-related information at such a site directly reflects the level of use or access of a particular subset of the totality of Resource Aliases by Client computers and users which connect to such a site.

Clients are used by information providers in order to make their resources accessible to users by registering Source and Resource Aliases and providing information associated with those Aliases. The Client sites retain a collection of recently used Resource Aliases and their related data including, but not limited to, the Addresses of the Electronic Resources associated with the Resource Aliases and descriptions of those Resources. The site is able to present the Resource Alias-related data to users, accept requests for retrieval of Resource Alias-related data, and invoke other software which may be resident on the same or other computers (such as World Wide Web browsers) in order to actually retrieve the Resources which the Resource Aliases represent. Client site requests for Resource Alias-related data are relayed to Local Servers which may either have such data cached (locally stored) or which may in turn request that information from the Central Registry or a Root Server on behalf of the Clients. The Client site may also accept requests for registration of new Resource or Source aliases on behalf of their users (who happen to be information providers) and submit those requests to the Central Registry (possibly via a Local Server). An important feature of the Client site is the ability to accept user requests for the "delivery" of the Electronic Resource associated with a Resource Alias. The satisfaction of this type of request may, take on several forms, including but not limited to: (1) invoking a software program such as a web browser, Gopher program or FTP (file transfer program) to access the document or resource, (2) sending a request to a Value Added Server to transmit the Resource in question to the user by postal service, electronic mail or facsimile, and (3) sending a request to a special server which provides direct voice telephone transmission of the information in the Resource to the user (either by human voice or by synthetic electronic voice). Information providers may elect to themselves pay for such delivery services of their Resources to the general

user community of the Network, or they may elect to provide the information on the condition that the recipient pay for the delivery services.

Value Added Servers provide various types of delivery services to clients, employing a number of electronic or non-electronic means, such as facsimile, telephone and postal service. Thus, a user may request of his Client site to have a text file associated with him/her bar Resource Alias sent to him/her by facsimile, or in printed form by postal service (mail), or by electronic mail, or (where relevant) read to him/her over the telephone. The Client site then relays such a request to the relevant Value Added Server site to have it fulfilled. Thus, a Resource Alias embodies a universal means of accessing the information associated with it.

This invention includes the individual modules at the aforementioned sites as well as the aggregate of the system which permits the individual sites to communicate with each other and act in concert as a system. The software imparts operating capacity to the sites which carry out the aforementioned primary functions as well as the additional functions delineated below. The various Server and Client sites mentioned are not intended to necessarily be special dedicated computers added to a Network in order to execute only the programs of this invention. Rather, they may be the computers already extant within the Network, augmented by the individual software programs and the aggregate of software of the invention in order to extend the utility of the Network.

6. Information Associated with a Resource Alias

A Resource Alias is a denotation of an Electronic Resource within a Network such as the Internet. Each Resource Alias has data associated with it, and is stored in the Central Registry. The data associated with a Resource Alias shall be termed the Resource Alias Record. Various other Server and Client sites also store certain Resource Aliases and their related Resource Alias Records. According to a preferred embodiment, a Resource Alias Records may include, but is not limited to, the following: the Address (or URL) of the Electronic Resource which the Resource Alias denotes, a description of the resource, a list of Resource Aliases of related Electronic Resources together with some minimal descriptive material, the last time and date when the Resource Alias Record was updated or changed, the time and date after which the Resource Alias and its record shall no longer valid, information about the owner or provider of the resource, and information about the format and/or type of document or resource in question (for example, text file, graphic file of a specific format or, web-page). Whenever a Local Server requests a Resource Alias from the Central Registry or a Root Server and whenever a Client computer requests such information from a Local Server, the aggregate of information delineated above (the Resource Alias Record) is transmitted to the requester.

7. Additional Functions of the Invention

In addition to the above-mentioned operations of the individual sites and of their aggregate systemic operation, the system according to the invention may provide additional functionalities. These are: (1) mechanisms for assuring that Resource Alias-related information is kept up-to-date in its various component systems; (2) mechanisms for assisting users in guessing or identifying Source and Resource Aliases associated with a particular information provider; and (3) mechanisms for providing users with the ability to invent and use even shorter mnemonic characters strings for representing and later accessing Electronic Resources.

The mechanism for keeping a Root Server up-to-date with respect to the Resource Aliases and Resource Alias Records it stores locally includes a periodic communication with the Central Registry, receiving updated Resource Aliases and Resource Alias Records, and updating its own database of such Resource Aliases and Resource Alias Records.

The mechanisms for keeping Local Servers up-to-date with respect to the Resource Aliases and Resource Alias Records they store locally may include the following: (1) A periodic inquiry in which a Local Server sends a list of its locally cached (stored) Resource Aliases to the Central Registry or to a Root Server, together with the times and dates when each of these Resource Aliases was last updated at that Server; in return the Central Registry or Root Server sends back a list of those Resource Aliases which are "out of date" (their associated data has since changed); the Local Server then deletes the out-of-date Resource Alias Records from its local cache. (2) Whenever a Client reports to a Local Server that it was not able to locate an Electronic Resource by using the Resource Alias Record Address it had been previously given, the Local Server will request an update of that information (the Resource Alias Record) from the Central Registry or from a Root Server and transmit it to the Client.

The mechanisms for keeping the Client computers up-to-date with respect to the Resource aliases and Resource alias Records they store locally may include the following: (1) a periodic inquiry in which a Client send a list of its locally cached (stored) Resource Aliases to its Local Server together with the times and dates when each of these Resource Aliases was last updated; in return the Local Server sends back a list of those Resource Aliases which are "out of date" (their associated data has since changed). The Local Server may need to, in turn, request such information from a Root Server or the Central Registry if it no longer retains some of the Resource Aliases or their Records. The Client computer then deletes the out-of-date Resource Alias Record from its local cache. (2) Whenever a Client attempts to use its locally stored Resource Alias Records (particularly, an Address or URL) for accessing an Electronic Resource and this attempt fails because the Address information is no longer correct, the (human) user is informed of this state of affairs and he/she is able to request updated information from the Local Server as above (by a simple action such as clicking on a button marked "Try Again").

The mechanism for registration of new Source Aliases will operate on a first come first serve basis wherein any information provider entity may request the use of a character string as a Source Alias for that entity. The character string is approved if (1) it is unique and (2) it conforms to the acceptable syntax for Source Aliases. Thereafter, the provider may choose any unique character string extension to formulate Resource Aliases for his/her resources so long as they conform syntactically and are indeed unique.

The mechanism for assisting users in guessing or identifying Source and Resource Aliases associated with a particular information provider may include the following: If a user (or the Client computer on behalf of the user) requests a Source Alias or Resource Alias which does not actually exist (is not on record at the Central Registry or at a Root Server), the Root Server or Central Registry shall attempt to match the character string submitted by the user with existing Source Alias and Resource Aliases, and it shall provide a list of possible candidate Source and Resource Aliases for the user. This communication may be performed via the Local Server. The user may then peruse the list of Aliases thus provided and possibly the information in the

Resource Alias Records associated with these Resource Aliases in order to determine if any of these denote the desired Electronic Resource. Various algorithms which deal with intelligently constructing candidate lists of Resource Aliases and/or Source aliases by locating Aliases which are visually and syntactically similar to the requested Alias.

The mechanism for providing users with the ability to invent and use even shorter mnemonic characters strings for representing and later accessing Electronic Resources may include the following: The Client site shall allow a user to create any characters sequence and associate that sequence with a Resource Alias or Source Alias stored (cached) at his Client computer. This sequence is termed a Nickname. Typing or otherwise entering or choosing this Nickname will have the same effect in the software as entering or choosing the Resource Alias which it denotes or represents. Thus, for example, a user may have cached the Resource Alias "General Motors/new car prices" and chosen a Nickname "GM$" for this Resource Alias—the use of these two character strings in calling up a Resource Alias Record or eliciting an Electronic Resource Address shall have the identical effect.

8. Use of Serial Numbers

Each Resource Alias may be associated with a unique serial number. This number is assigned by the Central Registry upon initial approval of the Resource Alias, and is used when transmitting lists of Resource aliases, wherein a list of serial numbers will be transmitted in place of the Resource Aliases in order to decrease the bandwidth required for such transmission. This is used when both parties involved in the transaction are able to ascertain the Resource Alias from its serial number.

The number space for the serial numbers is sufficiently large so that there is no need to reuse a serial number even if the Resource Alias with which it was originally associated is no longer in use. Serial numbers are assigned sequentially within the number space by the Central Registry Server.

9. Scope and Scale of the Invention

This invention may be implemented in networks and distributed systems of varying scope and scale. The various sites described, including Central Registry, Root Server, Local Server, Client and Value Added Server, may well be located at separate sites and separate computers in a large distributed system. However, the various servers and clients described are actually processes running on various computers and interacting so as to form one distributed system. In smaller implementations, some or all of the various servers described may actually be processes running on the same computer. In the most degenerate case, there would be a single Central Registry computer which would directly serve Client computers, providing Central Registry services and Local Server services simultaneously. It could also provide the various Value Added Services which might otherwise be provided by processes executing on separate networked computers. Root Servers are not actually needed in small scale systems wherein the Central Registry Services provided by the Central Registry Server suffices to respond adequately to all requests in a timely manner. Thus, the various servers are actually cooperating processes, and these may cooperate within a single site or computer or across distributed or networked systems. The only proviso is that the Client processes located in disparate computers serving their human users and communicating with a Server providing the services described as those provided by Central Registry, Local Server and Value Added Server.

BRIEF DESCRIPTION OF THE DRAWINGS

The drawings described here relate to a generic and specific implementation of the invention.

FIG. **1** shows a schematic view of an embodiment of the invention;

FIG. **2** shows a small scale implementation;

FIG. **3** shows a resource alias database;

FIG. **4** shows a client nickname and Resource Alias Cache;

FIG. **5** shows the flow of handling of user requests for Resource alias data; and

FIG. **6** shows dual implementation of a preferred embodiment.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. **1** depicts a systemic view of the invention, in which the various regions are Servers and/or Clients grouped according to their systemic function. The Central Services region **101** includes a Central Registry server **102** set of Root Servers **103** which provide a central repository for all Resource Aliases and their associated records; the servers in this region transmit the record of a given Resource Alias on request, provide lists of proximate Resource Aliases when a request is made for the record associated with a character sequence which is not a valid Resource Alias, and accept new requests for Resource Alias registry wherein they add the requested Resource Alias and its record to the repository. The Local Services region **104** includes Clients **105** which may require various services; in the full large scale implementation, these services are mediated, for each Client, by a Local Server **106** which also caches such Resource Aliases and associated records which have a high local demand. The Value Added Services region **107** includes a set of servers **108** which provide delivery of Electronic Resources on behalf of their owners or distributors by various electronic and nonelectronic means on request of Clients or their intermediaries.

FIG. **2** depicts a small scale form of the invention in which the various servers are co-located in a single machine. In this case, the Local Server cache of Resource Aliases related records is not necessary, as the Local Server process may directly access the database of the Central Registry Server. Furthermore (as is also the case depicted in FIG. **1**), the various servers should be thought of as cooperating processes (programs which are concurrently executing) which, in the case of FIG. **2**, happen to be co-resident in the same computer. Various intermediate configurations are also possible, in which some but not all of the server processes co-located at the same site or computer. In this figure (FIG. **2**), the various cooperating software components of the invention are depicted as circles and the sites or computers in which they are resident and executing are depicted as rectangles. It is expected that the software of this invention will generally be co-located with other software and will share the resources of their common host machine.

The server computer or site **201** may be a network server and may include the central registry server **202**, and the Value Added Server **203**, each communicating with and working in concert with the Local Server **204**. The system may also include a plurality of client computers **205**, each including the client functions and features **206**. The clients illustrated in FIG. **2** correspond to the Clients **105** illustrated in FIG. **1**.

FIG. **3** illustrates the database **301** of Resource Aliases **302** and their associated serial or sequence numbers **303** and Records **304** maintained by the Central Registry Server, a Root Server or a Local Server **305**. While this database and

its access structure are not identically used in these types of servers, the databases do have in common the following: (1) they are indexed for efficient retrieval of an item (a Resource Alias, its serial number and its Record) when requested either by supplying its Resource Alias or its serial number; (2) they may concurrently serve several requesters seeking retrieval of Resource Alias Records; and (3) they may alter, delete or add items with minimum interference to retrieval requests—that is, there is no need to halt retrieval services while update occurs. The Local Server database is its cache, containing only certain of the Resource Aliases and their Records, while the Central Registry maintains the entire collection of Resource Aliases in its database. A Root Server may be slightly out-of-phase with the Central Registry Server, and contain the entire Resource Alias collection current through the last update interaction with the Central Registry. FIG. **3** is not intended to limit the physical or logical structure of the database, as its structure may employ any accepted practice in structuring of data for efficient concurrent retrieval and rapid update.

FIG. **4** depicts the cache (local store) of Resource Aliases which the Client maintains, together with the Nicknames which represent some subset of the Resource Aliases. This illustration is not meant to represent the actual structure of the Client database, but rather the logical association of Resource Aliases, their serial numbers and their Records, and the association of Nicknames to Resource Aliases. The Nickname space **401** and Resource alias space **402** will act as a single space when a user requests a Resource Alias Record or attempts to access the Electronic Resource associated with a Resource Alias. The dual space is searched for the Resource Alias and/or Nickname, and the appropriate action is taken if it is found (or a request is submitted to the Local Server if it is not found). In actual implementation, *2a* any of various methods for storing Nicknames. Resource Aliases and Serial Numbers may be used so that the search and update mechanisms are appropriately efficient in the Client, such as serial search, hash coding, and/or inverted lists.

The Client Nicknames **401** may be associated with Resource Aliases **402** or Resource Alias serial numbers **403**. The Resource Alias and serial numbers are associated with the remaining fields of the Resource Alias Record **404**.

FIG. **5** illustrates the procedure associated with a user request for the Record associated with a Nickname or Resource Alias. This Record may be used for perusal on screen or for request for the delivery of an Electronic Resource associated with the Nickname or Resource Alias. The illustration separates the processing of the requests into logical regions, processed by different (cooperating) processes. The Client process **501** is depicted as first searching its own cache via the Nickname or Resource Alias space for the record. That failing, the Client **501** requests the Record from its Local Server **502**. If the Local Server **502** does not have that data cached, it requests it of the Central Registry or of a Root Server **503**. If the Record is located in any of those places, the process which finds it sends the Record back, and each of the processes receiving it caches it; finally the Record is displayed for the User. If the Resource Alias is not found anywhere, it is not a valid Resource alias, and the Root Server or Central Registry **503** formulates a reply in which a list of proximate Resource Aliases is returned. The Client process **501** displays this list to the user so he/she may visually peruse the list and choose likely candidates for further requests.

A user request **504** for an address is initiated by invoking the request process **505** through the submission of a character sequence of a purported Resource Alias **506**. The Client process includes an inquiry into whether the sequence constitutes a Nickname **507**. If so, the client process will retrieve the Resource Alias associated with the Nickname **508**. If the sequence is not a Nickname, the Client process inquires into whether the sequence is a cached Resource Alias in the local Client Server **509**. If it is determined that the sequence is a cached Resource Alias or, upon retrieval of a Resource Alias associated with a Nickname, the Client will then retrieve the Resource Alias record **510**. In the event the character sequence is not a cached Resource Alias, the Client will send the character sequence to the local server and request a corresponding Resource Alias record at **511**. Upon receipt of such a request, the local server will query its cached Resource Aliases at **512**. If the sequence is a valid cached Resource Alias, the local service will retrieve the corresponding Resource Alias record at **513** and transmit acknowledgment of the inquiry and the Resource Alias record **514** back to the Client **501**. If the local server finds that the sequence is not a valid cached Resource Alias at **512**, it forwards the character sequence to the Central Registry or a Root Server **503**. The Root Server then ascertains whether the sequence is a valid Resource Alias at **515**. If so, the central registry or Root Server **503** retrieves the corresponding Resource Alias record and transmits the record with a Resource Alias acknowledgment **516** back to the local server **502**. The local server will then store the Resource Alias and its corresponding record at **517** and transmit an acknowledgment along with the Resource Alias record at **514** back to the Client **501**. Upon receipt of a Resource Alias and corresponding record from the local server **502**, the Client **501** will store the Resource Alias and its corresponding record in the local Client cache **518**. Upon confirmation of the validity of a character sequence as a Resource Alias at either step **510** or **518**, the Client will display the Resource Alias record to the user at **519** and await further action. The request process ends at step **520**. The further action may include the establishment of a Nickname for the returned Resource Alias or submission of an additional character sequence in connection with an address request. The system according to the invention may be linked with other operating modules which may use the address contained in the Resource Alias record to request a copy of the resource. At this point, any of the previously discussed value added services may also be invoked.

In the event that the local server **502** submits a character sequence purported to be a Resource Alias to the Central Registry or Root Server **503** and it is determined at **515** that the sequence is not a valid Resource Alias, the Central Registry or Root Server **503** may create a list of proximate Resource Aliases for transmission to the local server at step **521**. The creation of the list of proximate Resource Aliases may be through some type of search which identifies Aliases which are syntactically and/or visually proximate to the submitted character sequence. The Central Registry or Root Server returns the list to the local server **502**, which receives the candidate Resource Alias list and retransmits the list to the Client **501** at step **522**. Upon receipt of the candidate Resource Alias list, the client informs the user that the submitted character sequence is not a valid Resource Alias and displays the candidate Resource Alias list at step **523** and awaits further action at step **520**.

FIG. **6** shows the preferred embodiment of this invention in a dual implementation, wherein the entire invention as described is implemented at two levels. The first is a universal implementation within the Internet and WWW community **601**, providing a global aliasing service which

may be used by any information provider with electronic resources available to Internet users. The second is one or more small scale implementation(s) **602** wherein a single Internet site and/or domain of one or more servers may provide an aliasing mechanism for denoting electronic resources for its immediate community of users. Thus, a user at or connected to such a site thus has available two disparate sets of denotations, one for global electronic resources of the entire Internet, and one for the local resources within his/her immediate community. The Client system **603** discerns between the two types of Resource Aliases (and their attendant Nicknames if such be used) by marking the first type "global" and the second type "local." The same Resource Alias may then be used in a global and local sense, wherein the local Resource Aliases "mask" global ones (are preferred during searches), but wherein the user may override this masking by requesting his/her Client system to seek out; one type or the other. The Local Server site **602** now acts as a Local Server for the global Resource Aliases **603**, caching them in the manner described above, while it acts as the Central Registry for the local Resource Aliases **604**, maintaining a database of all these Resource Aliases and their Records, and providing update, deletion and insertion services for local Resource Aliases and their records.

FIG. **6** depicts the organization of the dual implementation of the preferred embodiment of this invention. In this implementation, the Client system **603** provides the user with the choice of whether to prefer the Local or the Wide Area or Global (Internet) interpretations of Resource Aliases. This preference guides the Local Server **604** to search for a Resource Alias or character string purported to be a Resource Alias first in its Local Registry **606** or first in the Global system **605**. If the preferred choice fails to match a Resource Alias, the secondary system (for that particular user) search is activated. Thus, each Client request is accompanied by system preference data. The Client User interface, in displaying lists of Resource Aliases or individual Resource Aliases and their associated Records, also displays whether that particular Resource Alias and Record are Local or Global (Wide Area or Internet). The Local Server **604** is advantageously linked with the Central Registry or Root Server and Value Added Services, **607**, of the Wide Area System.

A fictional example of such usage is one wherein the U.S. Department of Agriculture registers a global Internet Source Alias "DOA" and an associated Resource Alias "DOA/Pathology" which provides information about animal or plant pathology resources, and wherein a hospital complex maintains an Internet domain which also serves as a Central Registry for local community Resource Aliases and registers a Source Alias "DOA" for information about "dead on arrival" and a Resource Alias "DOA/Pathology" for information about its pathology information database for DOD. A user of this community would elicit the local Resource Alias Record in response to a request for "DOD/Pathology" but could override this response by requesting global Resource Aliases. The local server would, in this case, request the Resource Alias Record from the Internet Central Registry on behalf of the Client and the User. Alternatively, the system could search the global, the local and the nickname caches for a sequence match and return all Resource Alias records corresponding to any match. The matching records would then be displayed for user selection or further action.

We claim:

1. An electronic resource denotation, request and delivery system within a network which shares information resources among its user community, comprising:

a central registry computer whose action is directed by software components,

one or more local server computers whose actions are directed by software components and linked to the central registry computer;

one or more client computers whose actions are directed by software components, and linked to a local server computer;

wherein the software components in these computers operate in concert as a distributed entity to allow client computers to denote resources with aliases that are unique across said server computers and said client computers, and further allow client computers to retrieve information corresponding to said aliases; and

wherein said aliases are maintained in at least said central registry computer and one or more of said local server computers.

2. The electronic resource denotation, request and delivery system of claim **1** further comprising:

computer memory associated with the central registry computer containing alias records of valid electronic resource aliases in said network and wherein said alias records contain electronic resource aliases and electronic resource addresses associated with said aliases; and

wherein the central registry's computer software transmits at least a portion of an alias record to a local server computer on request.

3. The electronic resource denotation, request and delivery system of claim **2**, further comprising:

one or more root server computers whose actions are directed by software components, associated with said central registry computer and linked to at least one of said local server computers; and

computer memory associated with the root server computers.

4. The electronic resource denotation, request and delivery system of claim **3**, wherein said computer memory associated with a root server computer contains copies of records stored in the computer memory associated with said central registry computer;

wherein a root server computer's software periodically requests the central registry computer for updates of all records that have been added, deleted or altered since the last update, and upon receipt of this update, stores the new or updated records in the computer memory associated with said root server computer; and

wherein the software of a root server directs said root server to respond to a request for information in a manner similar to the central registry server.

5. The electronic resource denotation, request and delivery system of claim **4**, wherein said software directs said root server computer to delete or invalidate records which correspond to said updated records or which have been superseded or deleted from the memory associated with said central registry computer.

6. The electronic resource denotation, request and delivery system of claim **3**, further comprising:

computer memory associated with a local server computer containing copies of one or more of the alias records; and

wherein the software periodically directs the local server computer to send a list of all the aliases of the records it currently retains together with the time and date of their last known update to a recipient computer,

wherein said recipient computer is the central registry computer or a root server computer, and wherein the software directs the recipient computer to return a list of retained aliases whose records have since changed; and wherein the software directs the local server computer to delete or invalidate the records associated with those aliases from its associated computer memory.

7. The electronic resource denotation, request and delivery system of claim 3, further comprising:

computer memory associated with the client computer containing copies of one or more of the alias records; and

wherein the software directs the client computers to accept requests for resource addresses by aliases from a user operating the client computer and to display the alias record corresponding an alias if said alias record is contained in the memory associated with the client computer; and if such an alias record is not present in the memory associated with the client computer to request that alias record from the local server so that it may be displayed or otherwise employed; and wherein the local server computer transmits that alias record if it is present in the computer memory associated with the local server computer; and wherein the local server computer requests the corresponding alias record from the central registry computer or a root server computer in order to fulfill a client computer request if that alias record is not present in the computer memory associated with said local server computer.

8. The electronic resource denotation, request and delivery system of claim 3 further comprising:

client computer software components which respond to a request by a user to retrieve an electronic resource associated with an alias by first retrieving the alias record corresponding to the alias from a local server computer if that alias record is not already stored in the client computer and then either (a) invoking further software components resident in the client computer capable of retrieving and displaying and/or storing the electronic resource, or (b) transmitting a resource delivery request to a value added server computer.

9. The electronic resource denotation request and delivery system of claim 8 wherein said resource delivery request specifies delivery by electronic mail or by facsimile or by postal service or by telephone.

10. The electronic resource denotation, request and delivery system of claim 3 further comprising:

client computer software components which periodically transmit a list of all resource aliases currently stored in said memory associated with said client computer together with times and dates the associated alias records were last updated to a local server computer; and

local server computer software components which in response to said periodically transmitted list, determines which alias records corresponding to the transmitted resource aliases are no longer current, by (a) checking the computer memory associated with the local server and, (b) requesting alias records from the central registry computer or a root server which are not contained in the memory associated with the local server computer and retaining alias records received in response to said request in the memory associated with the local server for fulfillment of the current and future requests; and wherein the client computer, upon receipt of the list of aliases which are no longer current, deletes corresponding alias records from its associated computer memory.

11. The electronic resource denotation, request and delivery system of claim 7 further comprising:

client computer software components which respond to a user request to retrieve an electronic resource associated with a character sequence which does not correspond to an alias stored in computer memory in the system by transmitting that character sequence to a local server computer, which in turn transmits it to the central registry computer in order to seek a corresponding alias record; wherein the central registry computer responds by acknowledging the request, and transmitting to the client computer, via the local server computer, a list of candidate aliases which approximate the character sequence; and wherein the client computer displays said list for user selection.

12. The electronic resource denotation, request and delivery system of claim 11, wherein the candidate aliases are proximate to the character sequence in that they all begin with the same character sequence or that they are the result of minor changes in the requested character sequence such as (a) omitting one or several characters in the sequence or (b) inserting one or several additional characters in the sequence or (c) altering several characters in the sequence.

13. The electronic resource denotation, request and delivery system of claim 2 further comprising:

client computer software components which respond to a user request to create a new alias for an electronic resource by eliciting from the user record information including an electronic resource address; wherein the client computer transmits the new alias record to the central registry computer; wherein the central registry computer either (a) approves the new alias and stores its record if the character sequence of the alias is valid and not in use; and wherein the central registry computer sends acknowledgement of acceptance to the client computer, or (b) denies approval of the new alias and transmits the reason for denial to the client computer; and wherein the client computer informs the user of the central registry computer's decision.

14. The electronic resource denotation, request and delivery system of claim 13 wherein said record information further comprises a resource format, text describing the resource, and other aliases whose corresponding resources are related to the resource.

15. The electronic resource denotation, request and delivery system of claim 2 further comprising:

client computer software components which respond to a user request to create a personal alias or nickname for an alias which is stored in associated computer memory by storing the personal alias and associating it with the resource alias; and wherein thereafter, in response to a request for the record corresponding to the personal alias or the request for the electronic resource associated with that personal alias, the software responds in the same manner as it would if the corresponding resource alias were used.

16. An electronic resource denotation, request and delivery system within a network which shares information resources among its user community, comprising:

a registry computer whose action is directed by software components;

one or more client computers whose actions are directed by software components, and linked to a registry computer; wherein the software components in these com-

puters operate in concert as a distributed entity to allow client computers to denote resources with aliases that are unique across said registry computer and said client computers, and further allow client computers to retrieve information corresponding to said aliases; a

wherein said aliases are maintained in at least said registry computer and one or more of said client computers.

17. The electronic resource denotation, request and delivery system of claim **16** further comprising:

computer memory associated with the registry computer containing alias records of valid electronic resource aliases in said network and wherein said alias records contain electronic resource aliases and electronic resource addresses associated with said aliases; and

wherein the registry's computer software transmits at least a portion of an alias record to a client computer on request.

18. The electronic resource denotation, request and delivery system of claim **17**, further comprising:

one or more mirror registry computers whose actions are directed by software components, associated with a central registry computer and linked to sat least one of said client computers; and

computer memory associated with the mirror registry computers.

19. The electronic resource denotation, request and delivery system of claim **18**, wherein said computer memory associated with a mirror registry computer contains copies of records stored in the computer memory associated with said registry computer;

wherein a mirror registry computer's software periodically requests the registry computer for updates of all records that have been added, deleted or altered since the last update, and upon receipt of this update, stores the new or updated records in the computer memory associated with said mirror registry computer; and

wherein the software of a mirror registry computer directs said mirror registry computer to respond to a request for information in a manner similar to the central registry server.

20. The electronic resource denotation, request and delivery system of claim **19**, wherein said software directs said mirror registry computer to delete or invalidate records which correspond to said updated records or which have been superseded or deleted from the memory associated with said central registry computer.

21. The electronic resource denotation, request and delivery system of claim **18**, further comprising:

computer memory associated with a client computer containing copies of one or more of the alias records; and

wherein the software periodically directs the client computer to send a list of all the aliases of the records it currently retains together with the time and date of their last known update to a recipient computer, wherein said recipient computer is the central registry computer or a mirror registry computer, and wherein the software directs the recipient computer to return a list of retained aliases whose records have since changed; and wherein the software directs the local server computer to delete or invalidate the records associated with those aliases from its associated computer memory.

22. The electronic resource denotation, request and delivery system of claim **18**, further comprising:

computer memory associated with the client computer containing copies of one or more of the alias records; and

wherein the software directs the client computers to accept requests for resource addresses by aliases from a user operating the client computer and to display the alias record corresponding an alias if said alias record is contained in the memory associated with the client computer; and if such an alias record is not present in the memory associated with the client computer to request that alias record from an associated registry computer so that it may be displayed or otherwise employed; and wherein the associated registry computer transmits that alias record if it is present in the computer memory associated with the associated registry computer.

23. The electronic resource denotation, request and delivery system of claim **18** further comprising:

client computer software components which respond to a request by a user to retrieve an electronic resource associated with an alias by first retrieving the alias record corresponding to the alias from an associated registry computer if that alias record is not already stored in the client computer and then either (a) invoking further software components resident in the client computer capable of retrieving and displaying and/or storing the electronic resource, or (b) transmitting a resource delivery request to a value added server computer.

24. The electronic resource denotation request and delivery system of claim **23** wherein said resource delivery request specifies delivery by electronic mail or by facsimile or by postal service or by telephone.

25. The electronic resource denotation, request and delivery system of claim **18** further comprising:

client computer software components which periodically transmit a list of all resource aliases currently stored in said memory associated with said client computer together with times and dates the associated alias records were last updated to a mirror registry computer; and

mirror registry computer software components which in response to said periodically transmitted list, determines which alias records corresponding to the transmitted resource aliases are no longer current, by (a) checking the computer memory associated with the mirror registry computer and, (b) requesting alias records from the registry computer which are not contained in the memory associated with the mirror registry computer and retaining alias records received in response to said request in the memory associated with the mirror registry computer for fulfillment of the current and future requests; and wherein the client computer, upon receipt of the list of aliases which are no longer current, deletes corresponding alias records from its associated computer memory.

26. The electronic resource denotation, request and delivery system of claim **22** further comprising:

client computer software components which respond to a user request to retrieve an electronic resource associated with a character sequence which does not correspond to an alias stored in computer memory in the system by transmitting that character sequence to a registry computer in order to seek a corresponding alias record; wherein the registry computer responds by acknowledging the request, and transmitting to the client computer a list of candidate aliases which approximate the character sequence; and wherein the client computer displays said list for user selection.

27. The electronic resource denotation, request and delivery system of claim **26**, wherein the candidate aliases are

proximate to the character sequence in that they all begin with the same character sequence or that they are the result of minor changes in the requested character sequence such as (a) omitting one or several characters in the sequence or (b) inserting one or several additional characters in the sequence or (c) altering several characters in the sequence.

28. The electronic resource denotation, request and delivery system of claim 17 further comprising:

client computer software components which respond to a user request to create a new alias for an electronic resource by eliciting from the user record information including an electronic resource address; wherein the client computer transmits the new alias record to a registry computer; wherein the registry computer either (a) approves the new alias and stores its record if the character sequence of the alias is valid and not in use; and wherein the registry computer sends acknowledgement of acceptance to the client computer, or (b) denies approval of the new alias and transmits the reason for denial to the client computer; and wherein the client computer informs the user of the registry computer's decision.

29. The electronic resource denotation, request and delivery system of claim 28 wherein said record information further comprises a resource format, text describing the resource, and other aliases whose corresponding resources are related to the resource.

30. The electronic resource denotation, request and delivery system of claim 17 further comprising:

client computer software components which respond to a user request to create a personal alias or nickname for an alias which is stored in associated computer memory by storing the personal alias and associating it with the resource alias; and wherein thereafter, in response to a request for the record corresponding to the personal alias or the request for the electronic resource associated with that personal alias, the software responds in the same manner as it would if the corresponding resource alias were used.

31. A method for aliasing electronic resources comprising the steps of:

entering an alias into a client computer and polling an associated memory for an electronic resource address associated with said alias;

using said alias to poll a memory associated with a local server for an address corresponding to said alias if said address is not in said client computer memory;

using said alias to poll a memory associated with a master server for an address corresponding to said alias if said address is not in said local server memory; and

returning the results of the polls to the client computer; where the results of the poll include an address coupled to the alias if a corresponding address is found.

* * * * *

US005790805A

United States Patent [19]

Bantum

[11] Patent Number: 5,790,805

[45] Date of Patent: Aug. 4, 1998

[54] **DISTRIBUTED TIMER SYNCHRONIZATION**

[75] Inventor: **Michael G. Bantum**, Sunnyvale, Calif.

[73] Assignee: **NCR Corporation**, Dayton, Ohio

[21] Appl. No.: **636,612**

[22] Filed: **Apr. 23, 1996**

[51] **Int. Cl.**[6] **G06F 15/163**

[52] **U.S. Cl.** **395/200.78**; 395/551; 364/271; 364/DIG. 1

[58] **Field of Search** ..... 395/200.78, 200.33, 395/551; 364/271, DIG. 1

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,531,185	7/1985	Halpern et al.	395/553
4,584,643	4/1986	Halpern et al.	395/200.78
4,827,401	5/1989	Hrustich et al.	395/551
5,434,997	7/1995	Landry et al.	395/182.1
5,535,217	7/1996	Cheung et al.	395/200.78
5,537,549	7/1996	Gee et al.	395/200.54
5,572,674	11/1996	Ernst	395/200.51

OTHER PUBLICATIONS

Mills, "Improved Algorithms for Synchronizing Computer Network Clocks," IEEE/ACM Transactions on Networking, vol. 3, No. 3, pp. 24455–254, Jun. 1995.

Bhonsle et al., "Virtual Process Clock: ATM Performance Measurements on UNIX Workstations," IEEE Global Telecommunications, pp. 893–897, Jun. 1995.

Couvet et al., "A Statistical Clock Synchronization Algorithm for Anisotropic Networks," Reliable Distributed Systems, 1991 10th Symposium, pp. 42–51, Mar. 1991.

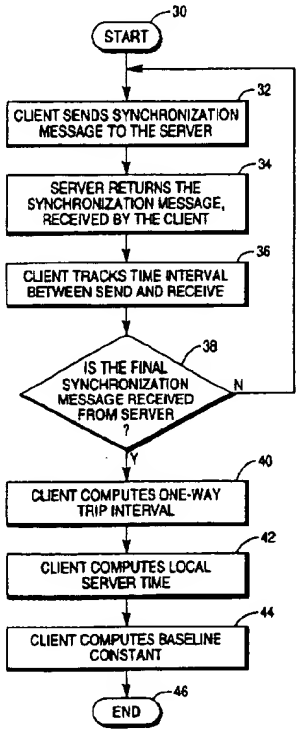
Primary Examiner—John E. Harrity

Attorney, Agent, or Firm—Merchant, Gould, Smith, Edell, Welter & Schmidt

[57] **ABSTRACT**

A method, apparatus, and article of manufacture for establishing an unlimited number of independent, client-based timers, synchronized with a timer kept on a central server, is disclosed. After forming a client-server connection, a client sends a synchronization message to a server. The client receives a return synchronization message from the server, and computes a round-trip interval time between sending and receiving, by sampling a local hardware clock. The sending and receiving of synchronization messages continues for a predetermined number of times, until the client receives a final synchronization message from the server, the final synchronization message including the current local server time. The client then calculates the average one-way trip interval and adds that value to the received current local server time, to provide the client with a reliable estimate of the local server time. By calculating the difference between the client's own local time and the calculated local server time, a constant is derived which can be used to calculate local server time for any and all future client local times.

30 Claims, 5 Drawing Sheets



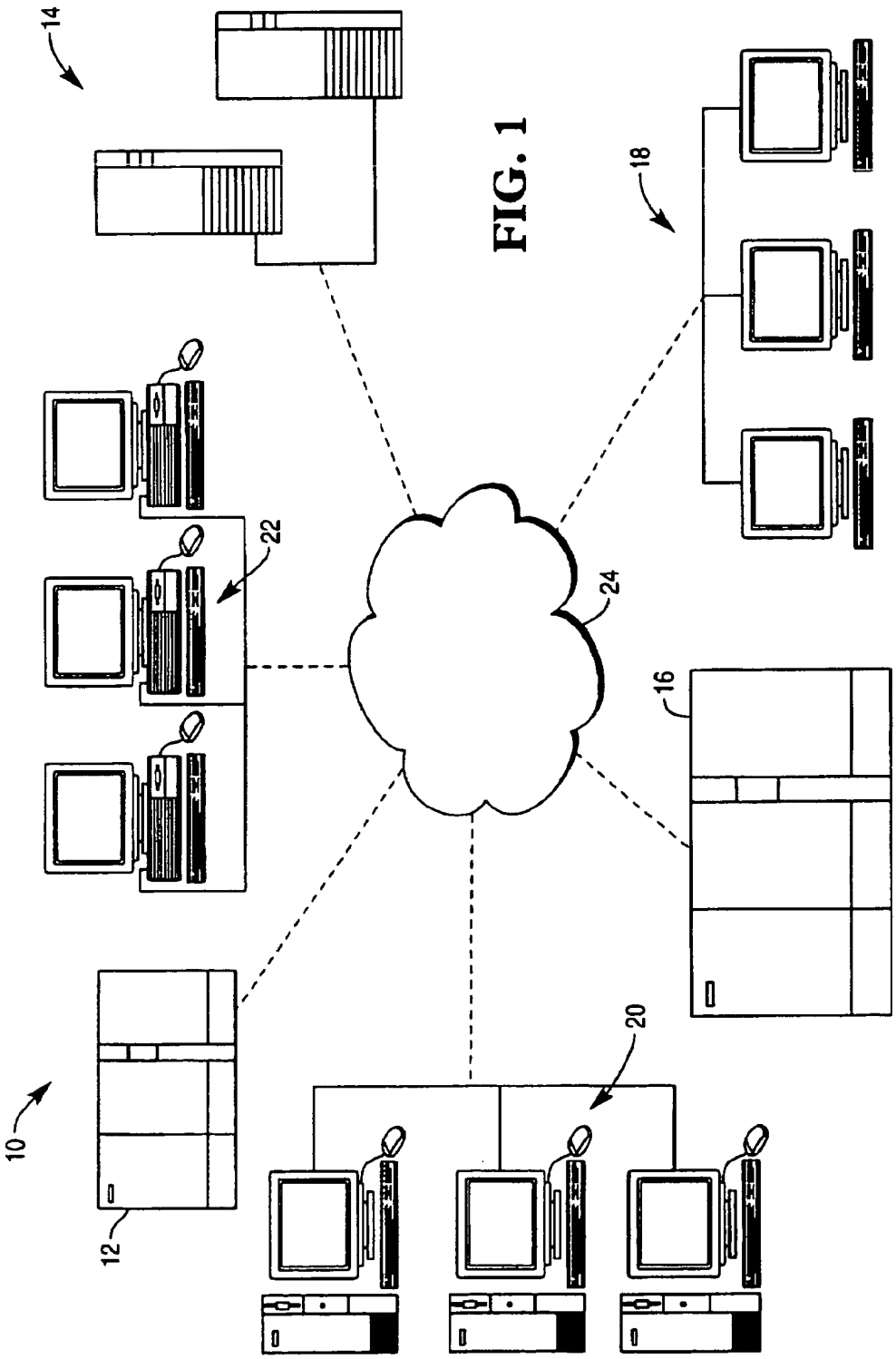


FIG. 1

FIG. 2A

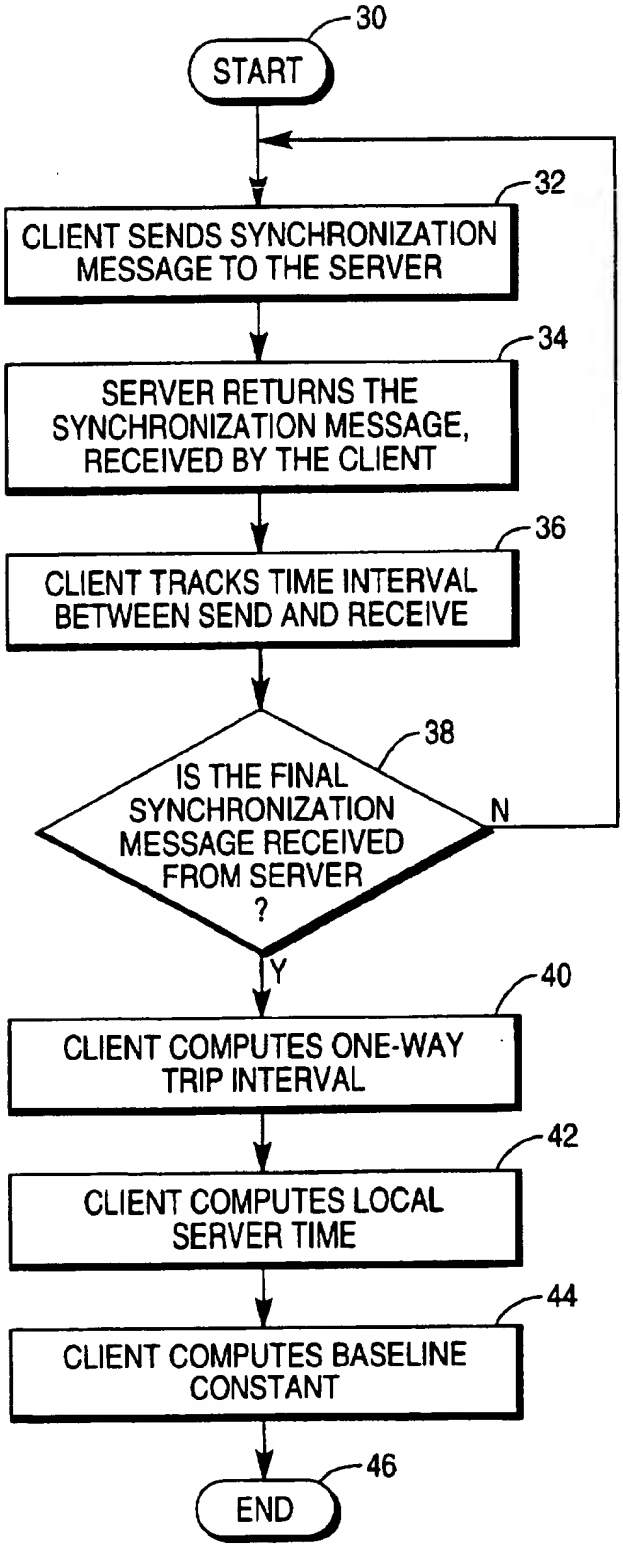

30
START
32
CLIENT SENDS SYNCHRONIZATION MESSAGE TO THE SERVER
34
SERVER RETURNS THE SYNCHRONIZATION MESSAGE, RECEIVED BY THE CLIENT
36
CLIENT TRACKS TIME INTERVAL BETWEEN SEND AND RECEIVE
38
IS THE FINAL SYNCHRONIZATION MESSAGE RECEIVED FROM SERVER ?
N
Y
40
CLIENT COMPUTES ONE-WAY TRIP INTERVAL
42
CLIENT COMPUTES LOCAL SERVER TIME
44
CLIENT COMPUTES BASELINE CONSTANT
46
END

FIG. 2B

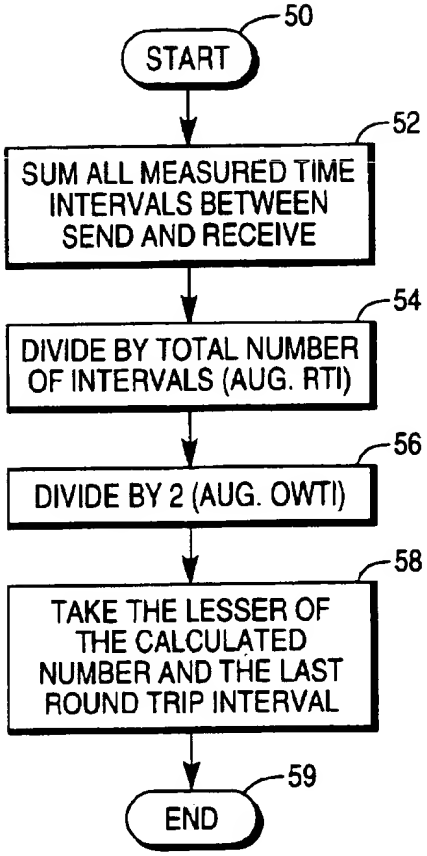


FIG. 2C

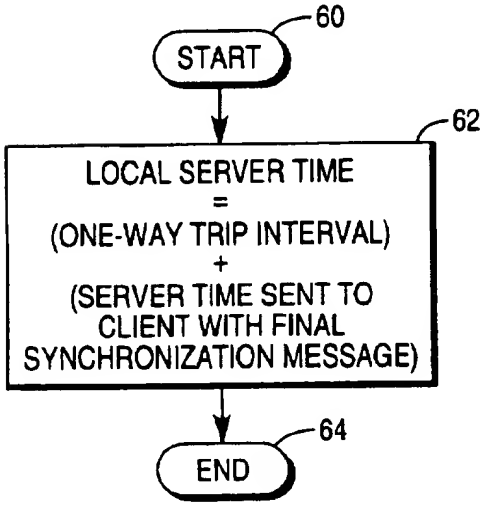


FIG. 2D

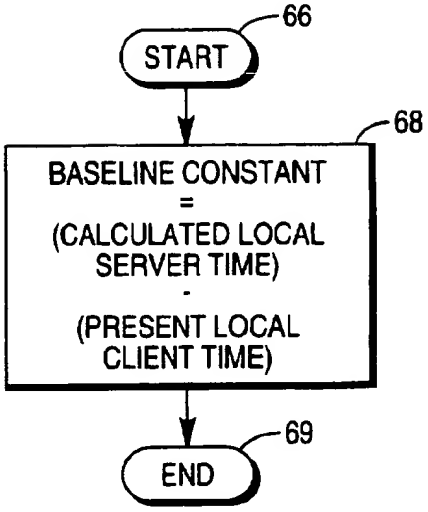


FIG. 2E

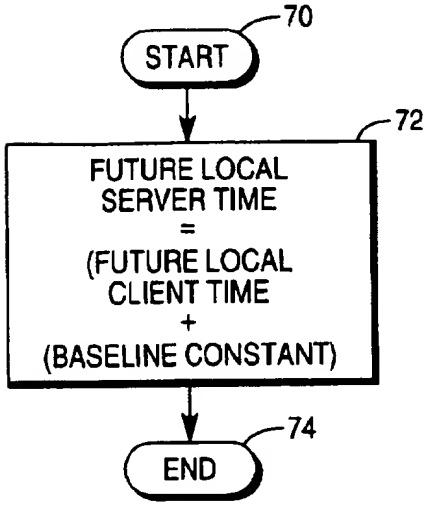
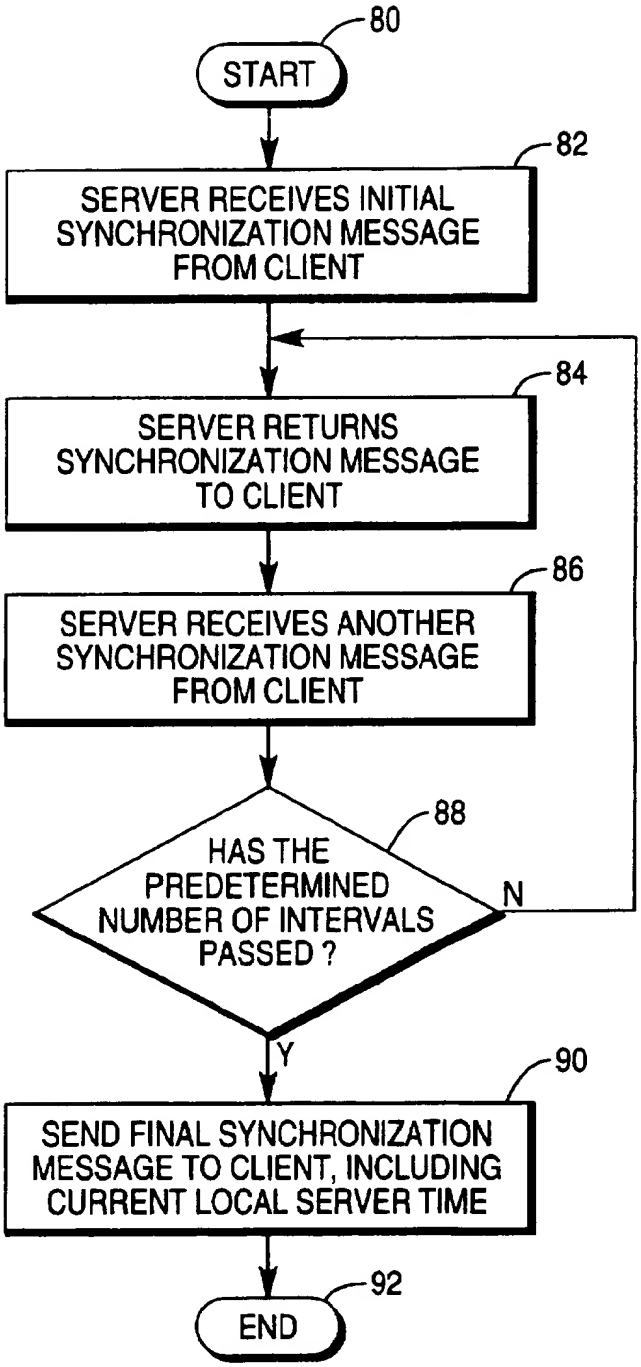


FIG. 3


80
START
82
SERVER RECEIVES INITIAL SYNCHRONIZATION MESSAGE FROM CLIENT
84
SERVER RETURNS SYNCHRONIZATION MESSAGE TO CLIENT
86
SERVER RECEIVES ANOTHER SYNCHRONIZATION MESSAGE FROM CLIENT
88
HAS THE PREDETERMINED NUMBER OF INTERVALS PASSED ?
N
Y
90
SEND FINAL SYNCHRONIZATION MESSAGE TO CLIENT, INCLUDING CURRENT LOCAL SERVER TIME
92
END

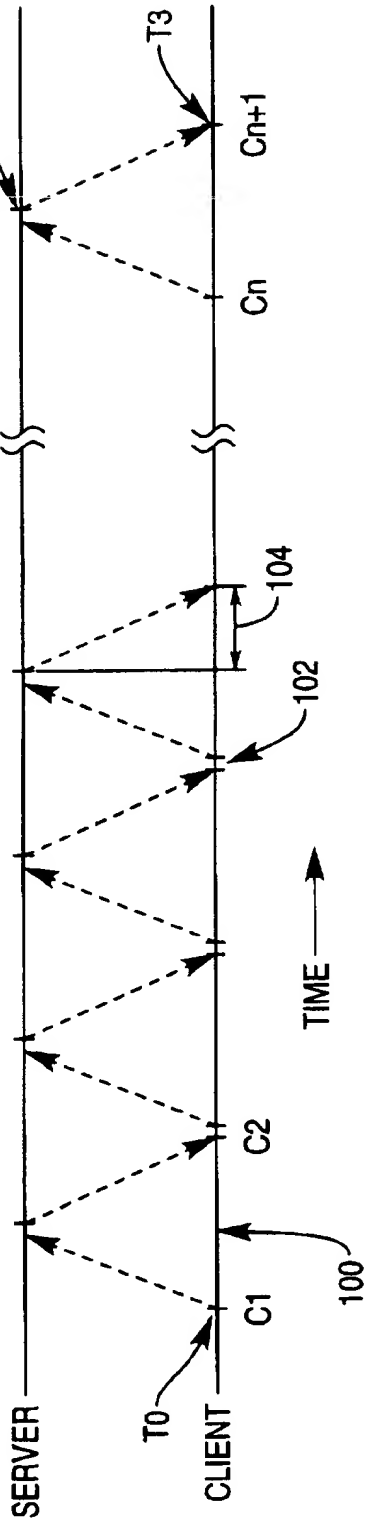


FIG. 4

DISTRIBUTED TIMER SYNCHRONIZATION

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates in general to client-server computer systems, and more particularly, to establishing a number of independent client-based software timers synchronized with a timer kept on a central server.

2. Description of Related Art

Computers are increasingly being used in the client-server configuration. In a client-server configuration, multiple computers are interconnected by a communication network. Certain computers are clients and other computers are servers. A client generates process requests, which are communicated to a server for processing. That is, a client generates requests and a server processes client requests. A particular computer can at times be a client and at other times a server.

One application of the client-server architecture is online transaction processing. Airline reservation and banking systems are classic examples of online transaction processing.

A principal advantage of the client-server architecture is the sharing of resources. Data, application programs, data storage devices, processing power, printers, communication subsystems, etc. can be shared. The client-server architecture also makes it possible to keep a centralized database, which is shared, as opposed to maintaining multiple copies of the same data with the overhead of insuring that the data remains consistent at all locations.

With continuing improvements in computer and communication technologies, the client-server architecture is being increasingly utilized. Computers can now be interconnected with local area networks and wide area networks, including wired telephone lines, cellular systems, satellite communication links, etc. The increased speed of communication networks that has been achieved have expanded the practical applications of client-server systems.

One significant problem associated with the client-server architecture is the lack of protocol for passively estimating one-way network transmittal times for purposes of network load analysis. Another problem is the lack of a reliable method for doing performance benchmarking of client-server applications.

A solution to these problems would be to maintain a software-based timer at each client, which is synchronized with the timer kept on a central server. However, the prior art discloses no adequate method or apparatus for accomplishing this important feature.

It can be seen then that there is a need for an efficient, reliable way to establish and maintain an unlimited number of local, software based timers at clients, each synchronized with the timer kept on a central server.

SUMMARY OF THE INVENTION

To overcome the limitations in the prior art described above, and to overcome other limitations that will become apparent upon reading and understanding the present specification, the present invention discloses a method, apparatus, and article of manufacture for establishing an unlimited number of independent, client-based timers, synchronized with a timer kept on a central server. After forming a client-server connection, a client sends a synchronization message to a server. The client receives a return synchronization message from the server, and computes a round-trip interval time between sending and receiving, by sampling a local hardware clock.

The sending and receiving of synchronization messages continues for a predetermined number of times, until the client receives a final synchronization message from the server, the final synchronization message including the current local server time. The client then calculates the average one-way trip interval and adds that value to the received current local server time, to provide the client with a reliable estimate of the approximate local server time at that instant.

In accordance with further aspects of the invention, in one embodiment the step of computing the average one-way trip interval includes using a function which imposes an upper bound on the average one-way trip interval, the upper bound equal to a last round-trip interval of the synchronization.

In accordance with further aspects of the invention, a baseline constant is computed by subtracting current local client time from the approximate local server time computed in the above method.

In accordance with still further aspects of the invention, the local server time can be computed at any future instant, simply by adding the local client time at the given future instant to the baseline constant computed using the above method. In this manner, a client-based timer is created and maintained, synchronized with a timer on a server. Further aspects of the invention will become apparent upon reading and understanding the present specification.

As will be appreciated from the foregoing brief summary of the invention, the object of the present invention is to establish and maintain a number of independent, client-based timers, synchronized with a timer kept on a central server. One advantage of achieving this objective is that these timers can be used to passively estimate one-way network transmittal times for purposes of network load analysis. Another advantage of achieving this objective is that these timers can be used for performance benchmarking of client-server applications. Yet another advantage of achieving this objective is that a number of derived network performance metrics can be generated with these timers. Further objects and advantages of this invention will become apparent upon reading and understanding the present specification.

These and various other advantages and features of novelty which characterize the invention are pointed out with particularity in the claims annexed hereto and form a part hereof. However, for a better understanding of the invention, its advantages, and the objects obtained by its use, reference should be made to the drawings which form a further part hereof, and to accompanying descriptive matter, in which there is illustrated and described specific examples of an apparatus in accordance with the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

Referring now to the drawings in which like reference numbers represent corresponding parts throughout:

FIG. 1 is a pictorial diagram of a client-server system with which the present invention can be used;

FIG. 2A is a flow chart depicting client application steps in accordance with this invention;

FIG. 2B is a flow chart depicting client computation of a one-way trip interval;

FIG. 2C is a flow chart depicting client computation of current local server time;

FIG. 2D is a flow chart depicting client computation of a baseline constant;

FIG. 2E is a flow chart depicting client computation of future local server time;

FIG. 3 is a flow chart depicting the steps performed by a server application according to this invention; and

FIG. 4 is a timing chart illustrating the interval timing when synchronizing the client and server.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

In the following description of the preferred embodiment, reference is made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration a specific embodiment in which the invention may be practiced. It is to be understood that other embodiments may be utilized and structural changes may be made without departing from the scope of the present invention.

FIG. 1 illustrates an exemplary client-server computer system **10** that could be used with the present invention. The system **10** includes: servers **12, 14** and **16**; clients **18, 20** and **22**; and a communication network(s) **24**, also known as a two-way network messaging service, which interconnects the servers and the clients. The clients illustrated are workstations **20**, personal computers **22**, and terminals **18**. Other clients, for example, laptop computers and personal digital assistants, are also possible. The servers are illustrated as mainframe and minicomputers; however, other computers, including smaller computers, could also take the role of the server. The communication network **24** can be comprised of many types of communication networks including local area networks and wide area networks, such as wired telephone lines, cellular telephone network and a satellite link. The communication network **24** can be made up of multiple networks of different types.

In the client-server system, the clients **18, 20, 22** generate requests that are processed by the servers **12, 14, 16**. The requests are transmitted from a client **18, 20, 22** to a server **12, 14, 16** via the communication network **24**. The servers **12, 14, 16** process the request and a response message is sent back over the communication network **24** to the client that made the request. For example, one of the personal computers **22** can send a request to the server **16** over the communication network **24**.

Software, running on both the clients **18, 20, 22**, and the servers **12, 14, 16**, together with communication hardware, makes communication between the clients and the servers possible. This client-server management software has been referred to as middleware. Various client-server management software packages are commercially available. For example, TOP END® is available from AT&T Global Information Solutions Co. Other companies packages include: Tuxedo, available from Novell Corp.; Encina, available from IBM; CICS/6000, available from IBM; Peer Logic; Noblenet; System Strategies; and others.

The servers **12, 14, 16** can work individually, each as a separate server. In that case, a client **18, 20, 22** would connect one of the servers **12, 14, 16** and any client request would be sent to and processed by that server. Alternatively, the servers **12, 14, 16** can work together to form a server system, which appears as a single server to the clients.

The present invention is preferably implemented using one or more computer programs. Generally, the computer programs are tangibly embodied in a computer-readable medium, e.g. floppy disk drives, hard disk drives, or CDROM disk drives. The computer programs may be loaded from the medium into the memory of the client **18, 20, 22** and the server **12, 14, 16**. The computer programs comprise instructions which, when read and executed by the client **18, 20, 22** and the server **12, 14, 16**, causes them to perform the steps necessary to execute the steps or elements of the present invention.

Those skilled in the art will recognize that the exemplary environment illustrated in FIG. **1** is not intended to limit the present invention. Indeed, those skilled in the art will recognize that other alternative hardware environments may be used without departing from the scope of the present invention.

FIG. 2A depicts the logic performed by the software components on clients **18, 20, 22** in one preferred embodiment of the invention. Synchronization begins with step **30** and continues as the client sends a synchronization message to the server **12, 14, 16** at step **32**. It is noted that, although this embodiment provides for client initiation of the synchronization, the invention is not so limited, and in fact the synchronization may be initiated at the server in further embodiments. After initiating the synchronization, the client at step **34** receives a return synchronization message from the server. The client tracks the time interval between the sending of the synchronization message and its receiving of the return synchronization message at step **36**. Next, at decision step **38**, the client checks to see if the final synchronization message has been received from the server. The final synchronization message is designated as such and includes the current local server time. If the final synchronization message has not been received, the client sends an additional synchronization message at step **32**. If the final synchronization message has been received from the server, the client continues to step **40** and computes the one-way trip interval.

Once the one-way trip interval has been computed, client-side timer synchronization continues, and the client computes the local server time at step **42**. In step **44**, the client computes a baseline constant, for use in computation of future local server time. Client-side timer synchronization then proceeds to end step **46** which completes the synchronization.

FIGS. 2B, 2C, and 2D further describe steps **40, 42**, and **44** of FIG. 2A respectively. Each flow chart in FIGS. **2B, 2C**, and **2D** provides a more detailed description of client computations during synchronization, in this embodiment of the invention.

Computation of the one-way trip interval (step **40** of FIG. 2A) is depicted in FIG. **2B**. The computation begins at start step **50**, and proceeds to step **52** where all time intervals between the send and receive, the intervals tracked in step **36** of FIG. 2A, are summed to determine a total. At step **54** the total number of time intervals are divided into the sum of all time intervals between sending and receiving, to determine the average round trip interval between send and receive. At step **56**, the average round trip interval is divided in half to determine the average one-way trip interval. At step **58** a minimum function is employed to provide an upper bound on the one-way trip interval. This is done by taking the lower of the calculated one-way trip interval and the last round trip time interval of the synchronization. At step **59**, the computation of the one-way trip interval ends.

Client-side computation of the local server time (step **42** of FIG. 2A) is depicted in FIG. **2C**. This computation begins with start step **60** and proceeds to step **62**, where the local server time is determined by summing the one-way trip interval time determined at step **40** of FIG. 2A, with the local server time sent to the client with the final synchronization message. Computation of local server time concludes at end step **64**.

Client computation of the baseline constant (step **44** of FIG. 2A) is depicted in FIG. 2D. The computation of the baseline constant begins at start step **66** and continues to step **68** where the baseline constant is calculated by subtracting the present local client time from the local server time calculated in step **42** of FIG. 2A. The calculation is concluded at end step **69**, which concludes the computation of the baseline constant.

FIG. 2E depicts all future computations by the client of the local server time, for all future local client times. Computation begins at start step **70** and proceeds to step **72**, where the future local server time is computed by adding the baseline constant to the future local client time. Computation is completed at end step **74**.

FIG. 3 depicts the logic performed by software components on servers **12**, **14**, **16** in one preferred embodiment of the invention. Server side timer synchronization begins at start step **80** and proceeds to step **82** where the server receives an initial synchronization message from the client. At step **84**, the server returns the synchronization message to the client, and at step **86** the server receives another synchronization message from the client. At decision step **88**, the server determines if it has received a predetermined number of synchronization messages from the client. If the predetermined number of synchronization messages has not been received from the client, the server proceeds to step **84** and returns another synchronization message to the client. If at step **88**, the server has received the predetermined number of synchronization messages from the client, the server sends a final synchronization message to the client, including the current local server time. This ends the server side timer synchronization at step **92**.

FIG. 4 is a timing diagram which depicts a passing of a synchronization message a predetermined number of times between a particular client and the server. In this case the client initiates the synchronization protocol. In this diagram, the predetermined number of times is equal to the constant n.

At initial server time **T0**, the client (at its local time **C1**) initiates synchronization by sending a message to the server, which receives the message at subsequent server time **S1**. The server receives and recognizes this client's synchronization message at subsequent server times **S1**, **S2**, . . . , Sn. The nth time the server receives the message (i.e., at server time Sn=**T1**), it ends the synchronization protocol by sending its current time back to the client which receives it at final server time **T3**, where Cn+1 represents the client's local time when it receives the last synchronization message from the server. The message's alternating network route between server and client is shown by dotted line arrows. Note that in general, **T0** is not equal to **C1** and **T3** is not equal to Cn+1, even though these times are the same relative to a global clock outside this system.

The first network round trip interval **100** computed by the client is the difference in its local send times or **C2**–**C1**. In general, any client round trip interval (RTI) can be adequately estimated as the difference, Ci+1–Ci. The processing time used by the client and server **102** (depicted as small gaps between the head of each arrow and the next labeled hash mark on each time line) is assumed to be negligible when compared with RTIs.

The spacing of the arrows in the diagram suggests that message one-way trip intervals (OWTI) **104** tend to be constant but for small, random variances, during the synchronization interval. When network conditions allow this, the accuracy of this method is best. The number of synchronization intervals is assumed to be few (i.e., n<10) for most applications, and it seems likely that point-to-point network conditions would hold fairly constant for short intervals. Also note that the following inequality must hold: T3–T1<Cn+1–Cn. This places an upper bound on the last OWTI of the synchronization message.

Several formulae provide guidance in performing the synchronization task. Assuming that OWTI tends towards an average over some number of network hops, the client may estimate the current time on the server, TS, as: TS=T1+min [Σ(Ci+1–Ci)/2n, Cn+1–Cn], the summation for all i such that 1<i<n+1.

This formula computes the average client RTI and divides it by two, adding this result to the server's current time, T1, as it last returned the synchronization message to the client. The last message RTI time, Cn+1–Cn, serves as an upper bound on this estimate and is enforced by the MIN operator in formula 1.

The synchronization by the client is completed by immediately subtracting its notion of local time, Cn+1, from TS above to derive a baseline value, TB (formula 2): TB=TS –Cn+1. Any future server time, FT>TB, corresponding to a local client time of Cn+j, can then be computed quickly and independently of the server by formula 3: FT+TB+Cn+j, for all j>0. The synchronization is thus completed for one client, and synchronized timers for additional clients can be established in the manner described above.

The foregoing description of the preferred embodiments of the invention has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed. Many modifications and variations are possible in light of the above teaching. It is intended that the scope of the invention be limited not with this detailed description, but rather by the claims appended hereto.

What is claimed is:

1. A method for synchronizing clocks between a client and a server in a computer-implemented network, comprising the steps of:

(a) exchanging synchronization messages between the client and the server a predetermined number of times, the synchronization message from server to client including a transmitted server time;

(b) computing at the client round-trip interval times for the exchanged synchronization messages;

(c) computing at the client a local server time based on the computed round-trip interval times and the transmitted server time; and

(d) offsetting the clock at the client to correspond to the computed local server time.

2. The method of claim 1 wherein the step (a) of exchanging synchronization messages includes the step of sending a synchronization message from a client to a server on a communication network.

3. The method of claim 2 wherein the step (a) of exchanging synchronization messages includes the step of receiving a return synchronization message from the server at the client on the communication network.

4. The method of claim 1 wherein the step (b) of computing round-trip interval times includes the step of computing at the client an individual round-trip interval time between the sending and receiving steps by sampling a local hardware clock.

5. The method of claim 4 wherein the step (c) of computing a local server time includes the step of computing at the client an average round-trip interval time by summing

the individual round-trip interval times and dividing by the predetermined number of times.

6. The method of claim 5 wherein the step (c) of computing the local server time includes the step of computing at the client a one-way trip interval time by dividing the average round-trip interval time in half.

7. The method of claim 6 wherein the step of computing the one-way trip interval time includes establishing an upper bound on the one-way trip interval time equal to a last round-trip interval time of the synchronization.

8. The method of claim 7 wherein the step (c) of computing the local server time includes the step of adding at the client the one-way trip interval time to the transmitted server time, the sum being equal to the computed local server time.

9. The method of claim 1 further including the step of computing at the client a baseline constant by subtracting current local client time from the computed local server time.

10. The method of claim 9 further including the step of independently computing at the client any future local server time by adding future local client time to the baseline constant.

11. A system for synchronizing clocks between a client and a server in a computer-implemented network, comprising:

(a) means for exchanging synchronization messages between the client and the server a predetermined number of times, the synchronization message from server to client including a transmitted server time;

(b) means for computing at the client round-trip interval times for the exchanged synchronization messages;

(c) means for computing at the client a local server time based on the computed round-trip interval times and the transmitted server time; and

(d) means for offsetting the clock at the client to correspond to the computed local server time.

12. The system of claim 11 wherein the means for exchanging synchronization messages (a) includes means for sending a synchronization message from a client to a server on a communication network.

13. The system of claim 12 wherein the means for exchanging synchronization messages (a) includes means for receiving a return synchronization message from the server at the client on the communication network.

14. The system of claim 11 wherein the means for computing round-trip interval times (b) includes means for computing at the client an individual round-trip interval time between the sending and receiving means by sampling a local hardware clock.

15. The system of claim 14 wherein the means for computing a local server time (c) includes means for computing at the client an average round-trip interval time by summing the individual round-trip interval times and dividing by the predetermined number of times.

16. The system of claim 15 wherein the means for computing the local server time (c) includes means for computing at the client a one-way trip interval time by dividing the average round-trip interval time in half.

17. The system of claim 16 wherein the means for computing the one-way trip interval time includes means for establishing an upper bound on the one-way trip interval time equal to a last round-trip interval time of the synchronization.

18. The system of claim 17 wherein the means for computing the local server time (c) includes means for adding at the client the one-way trip interval time to the transmitted server time, the sum being equal to the computed local server time.

19. The system of claim 11 further including means for computing at the client a baseline constant by subtracting current local client time from the computed local server time.

20. The system of claim 19 further including means for independently computing at the client any future local server time by adding future local client time to the baseline constant.

21. One or more program storage devices readable by a computer having a memory and coupled to a data storage device, each of the program storage devices tangibly embodying one or more programs of instructions executable by a computer to perform method steps for synchronizing clocks between a client and a server in a computer-implemented network, the method comprising the steps of:

(a) exchanging synchronization messages between the client and the server a predetermined number of times, the synchronization message from server to client including a transmitted server time;

(b) computing at the client round-trip interval times for the exchanged synchronization messages;

(c) computing at the client a local server time based on the computed round-trip interval times and the transmitted server time; and

(d) offsetting the clock at the client to correspond to the computed local server time.

22. The method of claim 21 wherein the step (a) of exchanging synchronization messages includes the step of sending a synchronization message from a client to a server on a communication network.

23. The method of claim 22 wherein the step (a) of exchanging synchronization messages includes the step of receiving a return synchronization message from the server at the client on the communication network.

24. The method of claim 21 wherein the step (b) of computing round-trip interval times includes the step of computing at the client an individual round-trip interval time between the sending and receiving steps by sampling a local hardware clock.

25. The method of claim 24 wherein the step (c) of computing a local server time includes the step of computing at the client an average round-trip interval time by summing the individual round-trip interval times and dividing by the predetermined number of times.

26. The method of claim 25 wherein the step (c) of computing the local server time includes the step of computing at the client a one-way trip interval time by dividing the average round-tip interval time in half.

27. The method of claim 26 wherein the step of computing the one-way trip interval time includes establishing an upper bound on the one-way tip interval time equal to a last round-trip interval time of the synchronization.

28. The method of claim 27 wherein the step (c) of computing the local server time includes the step of adding at the client the one-way trip interval time to the transmitted server time, the sum being equal to the computed local server time.

29. The method of claim 21 further including the step of computing at the client a baseline constant by subtracting current local client time from the computed local server time.

30. The method of claim 29 further including the step of independently computing at the client any future local server time by adding future local client time to the baseline constant.

* * * * *

US006178443B1

(12) **United States Patent**
Lin

(10) **Patent No.: US 6,178,443 B1**
(45) **Date of Patent: Jan. 23, 2001**

(54) **METHOD AND APPARATUS FOR PROPAGATING USER PREFERENCES ACROSS MULTIPLE COMPUTER ENVIRONMENTS**

(75) Inventor: **Chichuan Michael Lin**, Chandler, AZ (US)

(73) Assignee: **Intel Corporation**, Santa Clara, CA (US)

(*) Notice: Under 35 U.S.C. 154(b), the term of this patent shall be extended for 0 days.

(21) Appl. No.: **08/772,221**

(22) Filed: **Dec. 20, 1996**

(51) **Int. Cl.**[7] **G06F 15/16**

(52) **U.S. Cl.** **709/208**; 709/202; 709/209; 709/227; 707/104; 707/201

(58) **Field of Search** ..... 395/200.31–200.33, 395/200.38–200.39, 200.55–200.59, 200 T; 707/10, 9, 200–201, 100–104; 709/200–203, 208–209, 227–228, 238–239

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,800,488 * 1/1989 Agrawal et al. 709/225
5,541,927 * 7/1996 Kristol et al. 370/408
5,630,116 * 5/1997 Takaya et al. 707/201
5,678,041 * 10/1997 Baker et al. 709/229
5,819,284 * 10/1998 Farber et al. 707/104

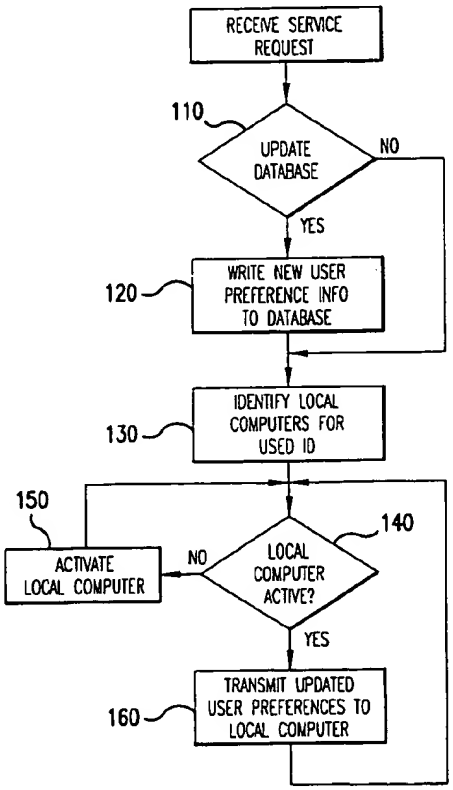
* cited by examiner

Primary Examiner—Ahmad F. Matar
Assistant Examiner—Bharat Barot
(74) *Attorney, Agent, or Firm*—Kenyon & Kenyon

(57) **ABSTRACT**

An embodiment of the present invention provides a method for propagating user preference information among a plurality of local computers coupled to a repository computer, where each local computer has one or more user-controllable parameters. According to the embodiment, the repository computer maintains central user preference information and transmits that information to a local computer. The local computer uses the transmitted user preference information to update one or more of its user-controllable parameters.

23 Claims, 3 Drawing Sheets



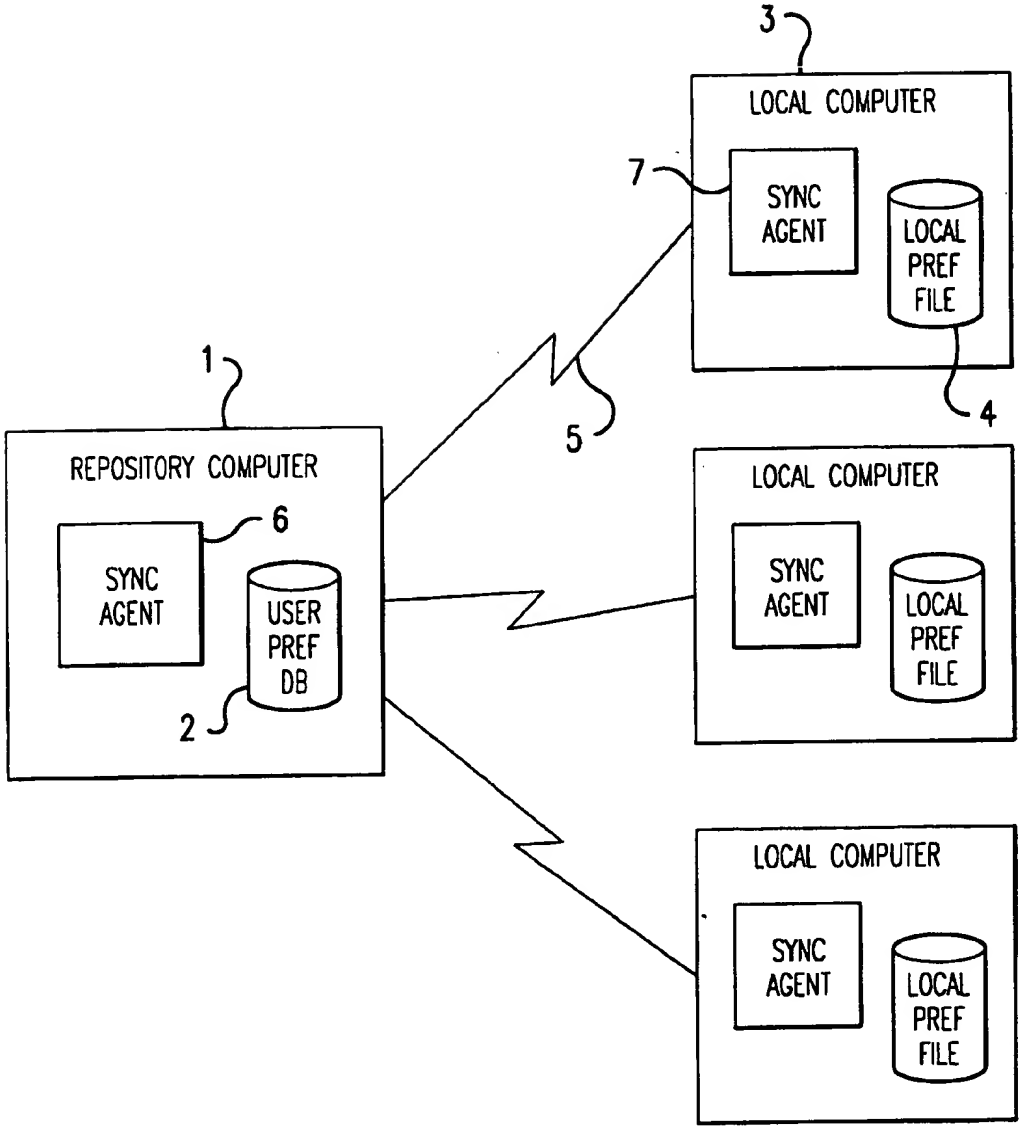

3
LOCAL COMPUTER
7
SYNC AGENT
LOCAL PREF FILE
4
1
5
REPOSITORY COMPUTER
6
SYNC AGENT
USER PREF DB
2
LOCAL COMPUTER
SYNC AGENT
LOCAL PREF FILE
LOCAL COMPUTER
SYNC AGENT
LOCAL PREF FILE

FIG.1

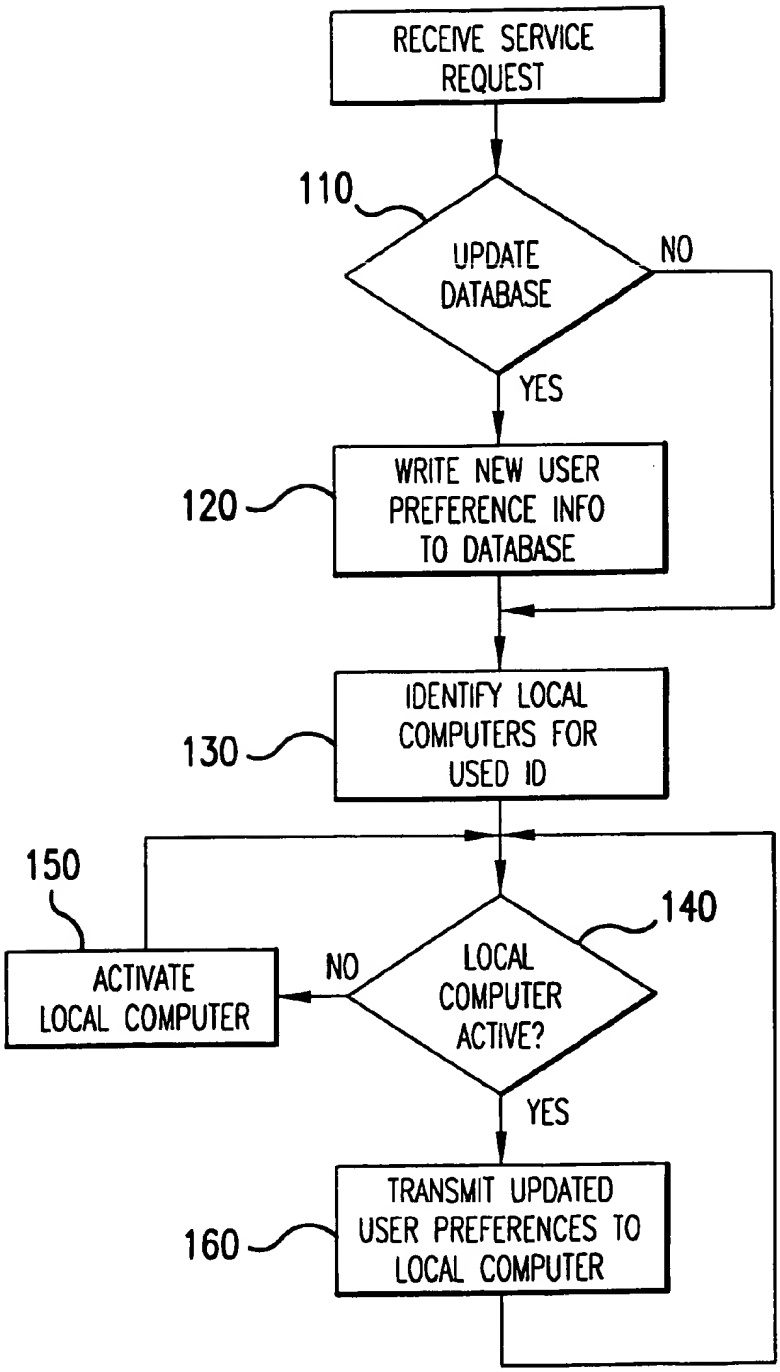

RECEIVE SERVICE REQUEST
110
UPDATE DATABASE
NO
YES
120
WRITE NEW USER PREFERENCE INFO TO DATABASE
130
IDENTIFY LOCAL COMPUTERS FOR USED ID
150
140
ACTIVATE LOCAL COMPUTER
NO
LOCAL COMPUTER ACTIVE?
YES
160
TRANSMIT UPDATED USER PREFERENCES TO LOCAL COMPUTER

FIG.2

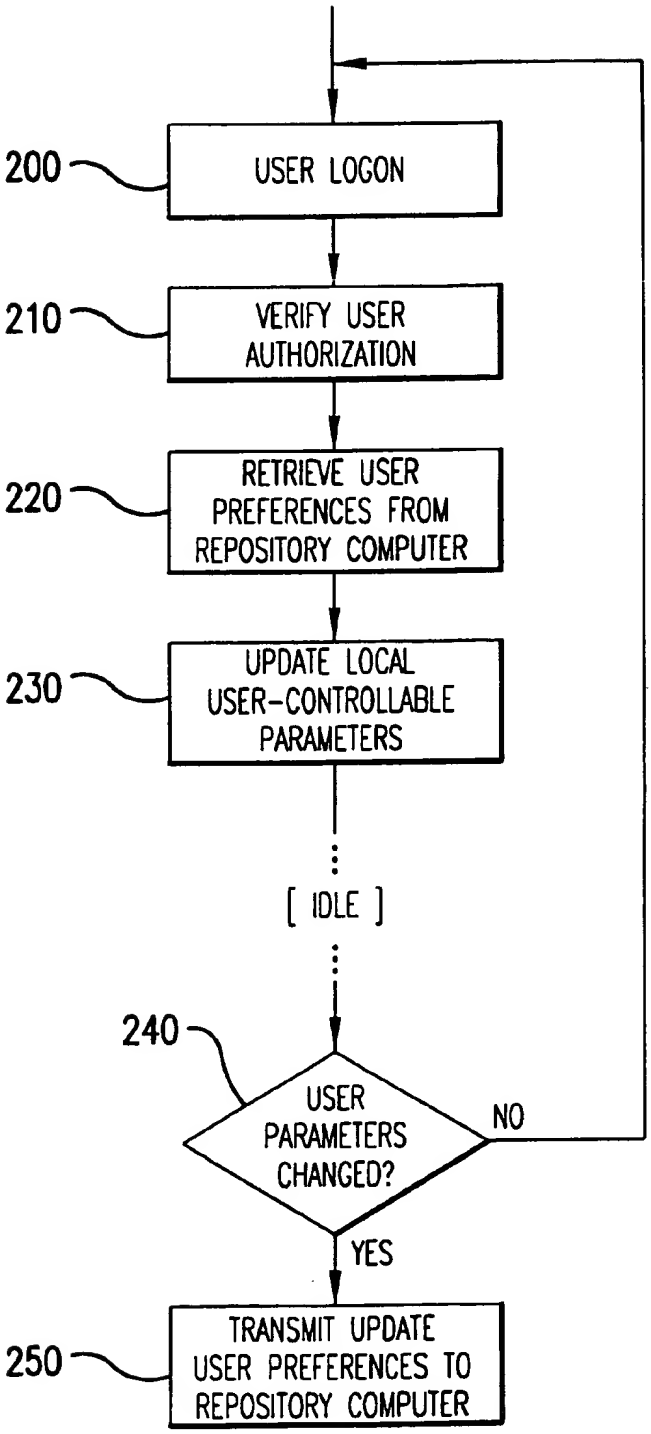

200
USER LOGON
210
VERIFY USER AUTHORIZATION
220
RETRIEVE USER PREFERENCES FROM REPOSITORY COMPUTER
230
UPDATE LOCAL USER-CONTROLLABLE PARAMETERS
[IDLE]
240
USER PARAMETERS CHANGED?
NO
YES
250
TRANSMIT UPDATE USER PREFERENCES TO REPOSITORY COMPUTER

FIG.3

METHOD AND APPARATUS FOR PROPAGATING USER PREFERENCES ACROSS MULTIPLE COMPUTER ENVIRONMENTS

BACKGROUND OF THE INVENTION

The present invention relates generally to the field of personal computing, and in particular to a method and apparatus for propagating user preferences across multiple computer environments. The invention has particular usefulness for users who routinely access a number of different computers, often at different physical locations, for work and/or personal purposes. The present invention can relieve such users from having to separately establish preferences at each such computer relating to, for example, keyboard and display settings, word processing software settings, Internet browser "bookmarks," and the like.

In today's increasingly computer-dependent society, it is becoming more and more common for people to have or use multiple computers. In a work environment, for example, a person may use a desktop personal computer (PC) in his or her office, another desktop or laptop PC in a development lab, and yet another laptop PC for use when traveling. In addition, this same person may have a desktop PC at home which he or she uses for both work-related and recreational activities. A common problem faced by these multi-computer users is that user-controllable system and software preferences are generally not configured in the same way across these various computers because of a variety of system and software complexities. This problem can cause untold frustration for users forced to continually reenter or change configurations to maintain some semblance of uniformity across multiple computer environments.

SUMMARY OF THE INVENTION

An embodiment of the present invention provides a method for propagating user preference information among a plurality of local computers coupled to a repository computer, where each local computer has one or more user-controllable parameters. According to the embodiment, the repository computer maintains central user preference information and transmits that information to a local computer. The local computer uses the transmitted user preference information to update one or more of its user-controllable parameters.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 illustrates schematically an embodiment of the present invention in which a plurality of local computers access a central repository of user preference information.

FIG. 2 describes processing performed by a repository computer according to an embodiment of the present invention.

FIG. 3 describes processing performed by a local computer according to an embodiment of the present invention.

DETAILED DESCRIPTION

The present invention provides a method and apparatus for propagating user preferences across multiple computer environments. The invention solves the problem of inconsistent configurations often faced by users who have or use multiple computers, eliminating the need to continually reenter or change configurations to maintain consistency.

Referring now to FIG. 1, an embodiment of the present invention includes a repository computer 1 capable of communicating with a plurality of local computers 3 over a communications link 5. Repository computer 1 may be any computer having sufficient memory and processing power to serve as a central repository for user preference information, including a standard desktop PC. Alternatively, repository computer 1 may be a dedicated network server machine. Other than the general requirements of a reasonable amount of available memory and processing power, the present invention does not depend upon any particular characteristics of repository computer 1.

Repository computer 1 includes a central user preference database 2 for maintaining user preference information for a plurality of system users. The stored user preference information may be, for example, preferences as to system configuration parameters, such as display resolution, keyboard speed and audio volume, or application-level parameters, such as default settings for word processing software or URL "bookmarks" for Internet browsers. All user preference information in central user preference database 2 is preferably associated with a unique userid identifying a system user with whom the information is associated.

In the embodiment of FIG. 1, local computer 3 may be, for example, any desktop or laptop PC capable of communicating with repository computer 1. Local computer 3 preferably includes a local user preference file 4 which, similar to central user preference database 2, maintains user-specific preference information. Here, however, the preference information would be limited to users of local computer 3, whereas central user preference database 2 may contain user preference information for a potentially much larger population of users.

Communications link 5 coupling repository computer 1 and local computers 3 may be any medium enabling computer-to-computer communications. Examples of suitable communications media include a local area network (LAN), such as a token ring or Fast Ethernet LAN, an Internet or intranet network, a POTS (Plain Old Telephone System) connection, a wireless connection and a satellite connection. The present invention is not dependent upon any particular medium for communications link 5, the sole criterion being the ability to carry user preference information and related data in some form from one computer to another.

Repository computer 1 includes a server-side synchronization agent 6 which provides functionality relating to the maintenance and distribution of central user preference information. Server-side synchronization agent 6 is preferably implemented as a software routine which is loaded whenever repository computer 1 is powered up, and which remains active until repository computer 1 is powered down. Where repository computer 1 includes, for example, a Windows™3.n/95 operating system, server-side synchronization agent 6 may be implemented as a dynamic link library (DLL) invoked automatically at startup. Similarly, in a DOS environment, server-side synchronization agent 6 may be an executable program initiated by the AUTOEXEC.BAT file. Other implementations are also possible, depending upon the characteristics of the particular operating environment of repository computer 1.

Server-side synchronization agent 6 preferably has read-write access to central user preference database 2. Central user preference database 2 may be implemented as a relational database accessible through SQL (Structured Query Language). Alternatively, central user preference database 2 may be some other type of data structure, such as a hierar-

chical database or a sequential file. For efficiency purposes, server-side synchronization agent **6** is preferably able to directly access user preference information for any given userid or other form of user identification.

Similar to repository computer **1**, local computer **3** includes a client-side synchronization agent **7** responsible for maintenance and use of user preference information. Client-side synchronization agent **7** also preferably runs whenever local computer **3** is powered on. Client-side synchronization agent **7** preferably has read-write access to local user preference file **4**. Local user preference file **4** may be a stand-alone data structure which client-side synchronization agent **7** uses as a local repository of user preference information, or local user preference file **4** may comprise a collection of data structures used by an operating system and/or application programs installed on local computer **3** to maintain user-controllable settings.

Turning now to functional features of the present invention, FIG. **2** and FIG. **3** describe a method for propagating user preference information to multiple computers according to an embodiment of the present invention. FIG. **2** describes the method from the perspective of a repository computer, while FIG. **3** describes the method from the perspective of a local computer. For illustration purposes, structural features are identified using the same reference numbers used in FIG. **1**; however, the method illustrated in FIG. **2** and FIG. **3** is not limited to that structural embodiment.

Referring now to FIG. **2**, server-side synchronization agent **6** loaded on repository computer **1** is activated upon receipt of a service request (Step **100**). A service request may be originated by a user of repository computer **1**, such as a request to maintain user preference information in central user preference database **12**. A service request may also be originated by local computer **3**, such as a request to download user preference information to local computer **3** or a request to update central user preference database **2** with new user preference information input at local computer **3**.

Server-side synchronization agent **6** analyzes the received service request to determine whether central user preference database **2** is to be updated (Step **110**). If so, server-side synchronization agent **6** preferably extracts a unique userid from the received request and uses that userid to call central user preference database **2** to retrieve any previously-stored user preference information for the userid. If entries exist in central user preference database **2** for the userid, server-side synchronization agent **6** updates those entries as appropriate based on the content of the received request, and writes the updated entries back to central user preference database **2**; if no such entries exist for the userid, server-side synchronization agent **6** formats new entries and writes those new entries to central user preference database **2** (Step **120**).

Once any necessary updates are made to central user preference database **2**, server-side synchronization agent **6** identifies which local computer(s) **3** are to receive updated user preference information as a result of the service request (Step **130**). Where, for example, the service request is a request from a particular local computer **3** for a download of the most-current user preference information, server-side synchronization agent **6** may transmit the appropriate information only to that local computer **3**. On the other hand, where the service request necessitated an update to central user preference database **2**, server-side synchronization agent **6** preferably transmits the updated user preference information to all of the computers associated with the userid in the service request. To this end, central user preference database **2** may include a table or other data structure associating userids with one or more local computers **3**. Such a table or other data structure also preferably includes address information for all such local computers **3**, such as a TCP/IP address or a LAN node ID, to enable server-side synchronization agent **6** to transmit information to such local computers **3**.

Where a local computer **3** other than the one that originated the service request is to receive updated user preference information, server-side synchronization agent **6** preferably determines whether the target local computer is activated (powered on) (Step **140**). If not, server-side synchronization agent **6** may transmit an appropriate command, using known technologies, to cause the target local computer **3** to turn itself on (Step **150**). Once it confirms the target computer is activated, server-side synchronization agent **6** may transmit the mostcurrent user preference information associated with the userid in the service request to local computer **3** (Step **160**).

To some degree, the nature of communications link **5** controls the manner in which server-side synchronization agent **6** is able to ensure that multiple local computers **3** associated with a given userid receive current user preference information. Where, for example, repository computer **1** and local computers **3** are all installed in a single network, server-side synchronization agent **6** may issue a broadcast-type call to simultaneously transmit current user preference information to all appropriate local computers **3**. In some embodiments, however, any given local computer **3** may be directly coupled to only one other computer, either repository computer **1** or another local computer **3**. In such a case, server-side synchronization agent **6** may propagate central user preference information in a token-passing manner. In other words, server-side synchronization agent **6** may transmit user preference information to a first local computer **3**, first local computer **3** could update its local user preferences and retransmit the information to a second local computer **3** coupled to first local computer **3**, and so on until all local computers **3** associated with the userid are updated. Yet another possibility is that server-side synchronization agent **6** is restricted to transmitting user preference information only to a local computer **3** which originates a service request.

Referring now to FIG. **3**, processing begins in client-side synchronization agent **7** when a new user logs on to local computer **3** (Step **200**). Where local computer **3** is accessible to multiple users, such as a computer in a development lab, it is desirable for client-side synchronization agent to include a security mechanism to ensure potentially-sensitive user preferences (or potentially-sensitive information accessible as a result of particular user preferences) are only available to authorized personnel. Accordingly, client-side synchronization agent **7** may require the new user to verify his or her authorization (Step **210**). This may be done, for example, using an assigned userid or password, fingerprint matching or voiceprint matching. Client-side synchronization agent **7** may provide default user preference information to be used in the event the new user lacks appropriate authorization.

Once a user is logged in and his or her authorization is verified, client-side synchronization agent **7** tries to retrieve the most current user preference information for the user from repository computer **1** (Step **220**). If repository computer **1** successfully returns the requested user preferences, client-side synchronization agent **7** implements those preferences in local computer **3** by, or example, updating system configuration files and application program preference files

(Step 230). If repository computer 1 fails to return the requested information, either because no such information exists in central user preference database 2 or because of some problem with communications link 5, client-side synchronization agent 7 may attempt to retrieve user preference information for the current user from local user preference file 4. If local user preference information is found, client-side synchronization agent 7 implements those preferences; otherwise, client-side synchronization agent 7 may implement a set of default user preferences.

Once appropriate user preferences are implemented, client-side synchronization agent 7 ideally enters an idle state until a predetermined event occurs to trigger further processing. For example, client-side synchronization agent 7 may idle until the current user logs off of local computer 3. At that time, client-side synchronization agent 7 may compare current system settings to those it originally established as a result of the previously-stored user preference information to determine whether any user preferences changed during the just-completed user session (Step 240). If so, client-side synchronization agent 7 transmits the locally-updated user preference information to repository computer 1 (Step 250), thereby enabling repository computer 1 to propagate the most-current user preferences to other local computers 3 associated with that user. Alternatively, client-side synchronization agent 7 may inform repository computer 1 of user preference updates on an ad hoc basis. For example, where a user changes settings in a particular application, such as adding new bookmarks in an Internet browser, client-side synchronization agent 7 may transmit the updates to repository computer 1 as soon as the user exits the application.

In the embodiment illustrated in FIG. 1, server-side synchronization agent 6 and client-side synchronization agent 7 are illustrated as distinct software modules. Nevertheless, to maximize flexibility and production efficiency, both agents may be incorporated as part of a single application program configurable to function either as a server-side or a client-side synchronization agent.

Server-side synchronization agent 6 and/or client-side synchronization agent 7 may be distributed as pre-loaded software (comprising a set of executable instructions) resident in a memory of a personal computer, such as the hard-disk of a desktop or laptop computer. Alternatively, the software may be distributed to users in the form of a user-installable program stored on any of a variety of portable media, including diskette and CD. Yet another possibility is that the software could be made available on a network server for downloading upon request by a user.

The foregoing is a detailed description of particular embodiments of the present invention as defined in the claims set forth below. The invention embraces all alternatives, modifications and variations that fall within the letter and spirit of the claims, as well as all equivalents of the claimed subject matter.

What is claimed is:

1. A method for propagating user preference information among a plurality of local computers, wherein each local computer is coupled to a repository computer and has one or more user-controllable parameters, said method comprising the steps of:

maintaining user preference information regarding hardware and/or software configurations corresponding to each of a plurality of users on the repository computer;

associating a user of a local computer with corresponding user preference information on the repository computer;

transmitting the corresponding user preference information to the local computer; and

updating one or more user-controllable parameters of the local computer according to the transmitted corresponding user preference information.

2. The method of claim 1, wherein the repository computer transmits the corresponding user preference information upon receipt of a service request from the local computer.

3. The method of claim 1, wherein the repository computer transmits the corresponding user preference information following an update to the corresponding user preference information.

4. The method of claim 3, wherein the repository computer ensures the local computer is activated prior to transmitting the corresponding user preference information.

5. The method of claim 1, wherein the corresponding user preference information transmitted by the repository computer is propagated to a plurality of local computers.

6. The method of claim 5, wherein the repository computer transmits the corresponding user preference information to each of the plurality of local computers.

7. The method of claim 5, wherein each of the plurality of local computers is capable of communicating with at least one other of the plurality of local computers, said method further comprising the step of transmitting the corresponding user preference information from a first local computer to a second local computer.

8. The method of claim 7, wherein the corresponding user preference information originally transmitted by the repository computer is passed among the plurality of local computers until all of the plurality of local computers have received the corresponding user preference information.

9. The method of claim 1, wherein the repository computer comprises one of the plurality of local computers.

10. The method of claim 1, wherein the repository computer comprises a server computer to which each of the plurality of local computers is coupled.

11. A computer system in which user preference information is propagated among a plurality of computers, said computer system comprising:

one or more local computers, each of said local computers including a local data store containing user preference information regarding hardware and/or software configurations, said local data store being used by said local computer to establish default settings for one or more user-controllable parameters; and

a repository computer coupled to said one or more local computers, said repository computer including a database containing user preference information corresponding to each of a plurality of users and program code for selectively transmitting user preference information to a local computer, wherein the transmitted user preference information corresponds to a particular user of the local computer.

12. The computer system of claim 11, wherein said one or more local computers further include program code for transmitting to said repository computer a service request for user preference information corresponding to a particular user of the local computer, and said repository computer further includes program code for transmitting requested user preference information to said one or more local computers in response to said service request.

13. The computer system of claim 11, wherein said repository computer further includes program code for automatically transmitting user preference information to a local computer following an update of said user preference information on said repository computer.

14. The computer system of claim 13, wherein said repository computer further includes program code for activating a local computer prior to transmitting said user preference information to said local computer.

15. The computer system of claim 11, wherein said repository computer comprises a server computer and each of said one or more local computers comprises a client computer, said server computer and said client computers being coupled to one another over a network.

16. An instruction set residing on a storage medium for propagating user-specific configuration information among a plurality of computers, said instruction set comprising instructions for:

maintaining a central database of user preference information corresponding to each of a plurality of users, wherein said user preference information comprises information relating to at least one of a software configuration and a hardware configuration for a user computer; and

transmitting user preference information to a computer associated with a selected one of said plurality of users.

17. The instruction set of claim 16, further comprising instructions for:

identifying one or more computers associated with the selected user;

transmitting user preference information to each of said identified computers following an update of said user preference information included in said central database.

18. The instruction set of claim 17, further comprising instructions for ensuring said one or more computers are activated prior to transmitting said user preference information.

19. The instruction set of claim 16, further comprising instructions for:

reading a communication from a remote computer to identify a user of said remote computer; and

retrieving user preference information corresponding to said identified user from said central database.

20. The instruction set of claim 16, wherein said storage medium comprises a permanent storage device installed in a computer.

21. An apparatus for propagating user preference information among a plurality of local computers, wherein each of the plurality of local computers includes a local data store containing user preference information regarding hardware and/or software configurations, the local data store being used by the local computer to establish settings for one or more user-controllable parameters, said apparatus being coupled to at least one of the plurality of local computers and including a database containing user preference information corresponding to each of a plurality of uniquely-identifiable users and program code for selectively transmitting user preference information to a local computer for updating the local data store of the local computer, said transmitted user preference information corresponding to an identified user of the local computer.

22. The apparatus of claim 21, further including program code for automatically transmitting user preference information to a local computer following an update of said user preference information included in the database.

23. The apparatus of claim 21, further including program code for activating a local computer prior to transmitting said user preference information to the local computer.

* * * * *

US006571276B1

(12) **United States Patent**
Burns et al.

(10) **Patent No.: US 6,571,276 B1**
(45) **Date of Patent: May 27, 2003**

(54) **SYSTEM FOR MANAGING ASSET ACCESS IN A DISTRIBUTED STORAGE SYSTEM**

(75) Inventors: **Randal Chilton Burns**, Sunnyvale, CA (US); **Atul Goel**, San Jose, CA (US); **Robert Michael Rees**, Los Gatos, CA (US)

(73) Assignee: **International Business Machines Corporation**, Armonk, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/510,784**

(22) Filed: **Feb. 23, 2000**

(51) **Int. Cl.**[7] **G06F 17/30**; G06F 11/30

(52) **U.S. Cl.** **709/212**; 713/190; 707/8; 709/224; 709/225

(58) **Field of Search** 709/212, 225, 709/224; 707/8, 9, 10; 713/190

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,887,204	A *	12/1989	Johnson et al.	364/200
4,965,719	A *	10/1990	Shoens et al.	364/200
5,193,188	A	3/1993	Franaszek et al.	395/650
5,202,971	A *	4/1993	Henson et al.	395/425
5,305,448	A	4/1994	Insalaco et al.	395/425
5,339,427	A *	8/1994	Elko et al.	395/725
5,490,270	A *	2/1996	Devarakonda et al.	395/600
5,551,046	A *	8/1996	Mohan et al.	395/800
5,615,373	A *	3/1997	Ho	395/726
5,867,577	A	2/1999	Patarin	380/23
5,956,712	A *	9/1999	Bennett et al.	707/8
6,199,141	B1 *	3/2001	Weinreb et al.	711/118
6,272,662	B1 *	8/2001	Jadav et al.	714/805
6,324,581	B1 *	11/2001	Xu et al.	709/229
6,405,315	B1 *	6/2002	Burns et al.	713/190
6,466,978	B1 *	10/2002	Mukherjee et al.	709/225

OTHER PUBLICATIONS

B. Walker, G. Popek, R. English, C. Kline, and G. Thiel. "The LOCUS Distributed Operating systems." In Proceedings of the 9th ACM Symposium on Operating Systems Principles, 1983.*

A. D. Birrel and R. M. Needham. "A Universal File Server." IEEE Transactions on Software Engineering. SE6(5), Sep. 1980.*

D. Walsh, B. Lyon, G. Sager, J. Chang, D. Goldberth, S. Kleiman, T. Lyon, R. Sandberg, and P. Weiss. Overview of the Sun Network File System. In Proceeding of the 1985 Winter Usenix Technical Conference, Jan. 1985.*

M. L. Kazar. "Synchronization and Caching Issues in the Andrew File System." In Proceeding of the USENIX Winter Technical Conference, Feb. 1988.*

M. Devarakonda, D. Kish, and A. Mohindra. "Recovery in the Calypso File System." ACM Transactions on Computer Systems. Aug. 1996.*

M. L. Kazar, B. W. Leverett, O. T. Anderson, V. Apostolides, B. A. Bottos, S. Chutani, C. F. Everhart, W. A. Mason, S. Tu, and R. Zayas. "Decorum File System Architectural Overview." In Proceedings of the Summer USENIX Conference, Jun. 1990.*

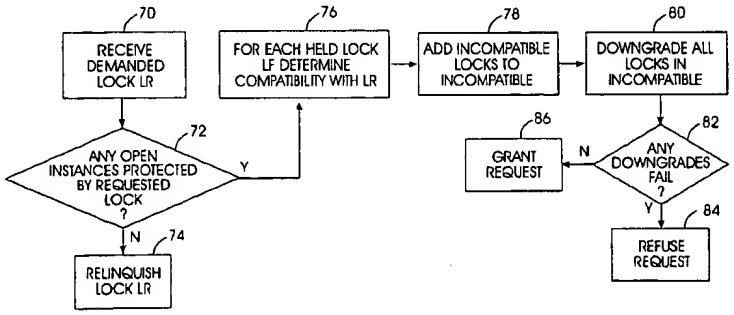
* cited by examiner

Primary Examiner—David Wiley
Assistant Examiner—Alina Boutah
(74) *Attorney, Agent, or Firm*—John L. Rogitz

(57) **ABSTRACT**

A system and method for managing access to assets in a distributed data storage system includes requesting, from a client computer, a semi-preemptible access lock from a server computer. The semi-preemptible lock, if granted, is held by the client computer as long as the server does not demand it back, with the client computer granting open instances under non-preemptible file locks for the asset to which the locks pertain as long as the client computer holds the semi-preemptible lock. When another client computer requests the semi-preemptible lock, the server can demand the lock from the holding client, which relinquishes the lock if no open instances are protected by the lock. Otherwise, the holding client computer first attempts to downgrade its lock to meet the request, and if compatibility is not achieved thereby, the holding client refuses to relinquish the lock.

10 Claims, 3 Drawing Sheets



CLIENT LOGIC FOR PROCESSING SEMI-PREEMPTIBLE LOCK DEMAND

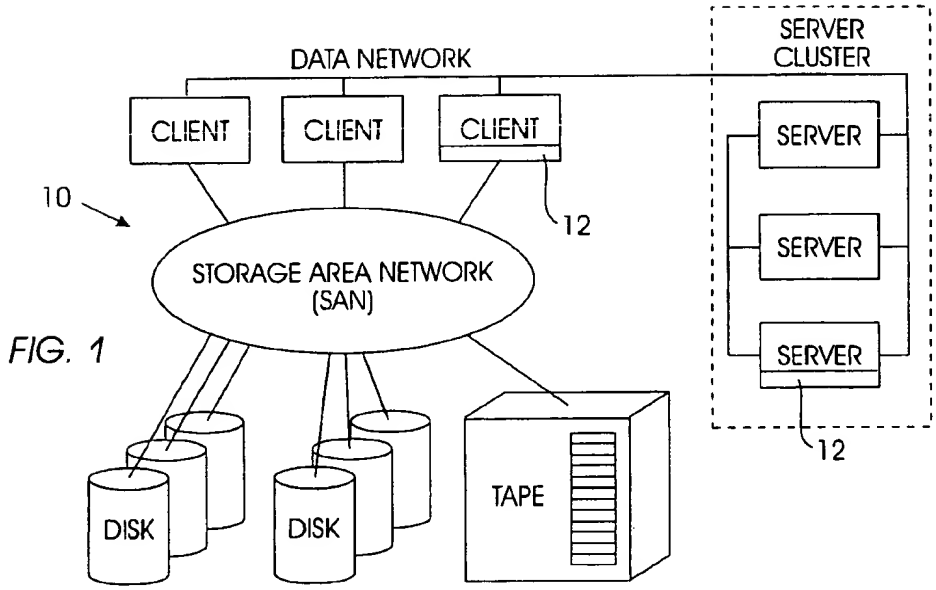


FIG. 1

14

LOCK	NAME	ACCESS	SHARING
M	METADATA	METADATA	METADATA, READ, WRITE
R	READ	METADATA, READ	METADATA, READ, WRITE
S	SHARED	METADATA, READ	METADATA, READ
W	WRITE	METADATA, READ, WRITE	METADATA, READ, WRITE
U	UPDATE	METADATA, READ, WRITE	METADATA, READ
X	EXCLUSIVE	METADATA, READ, WRITE	METADATA

FIG. 2

16

	M	R	W	S	U	X
M	✓	✓	✓	✓	✓	✓
R	✓	✓	✓	✓	✓	
W	✓	✓	✓			
S	✓	✓		✓		
U	✓	✓				
X	✓					

FIG. 3

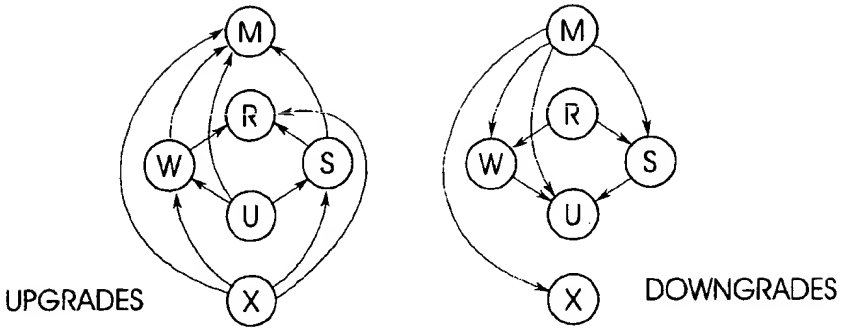


FIG. 4

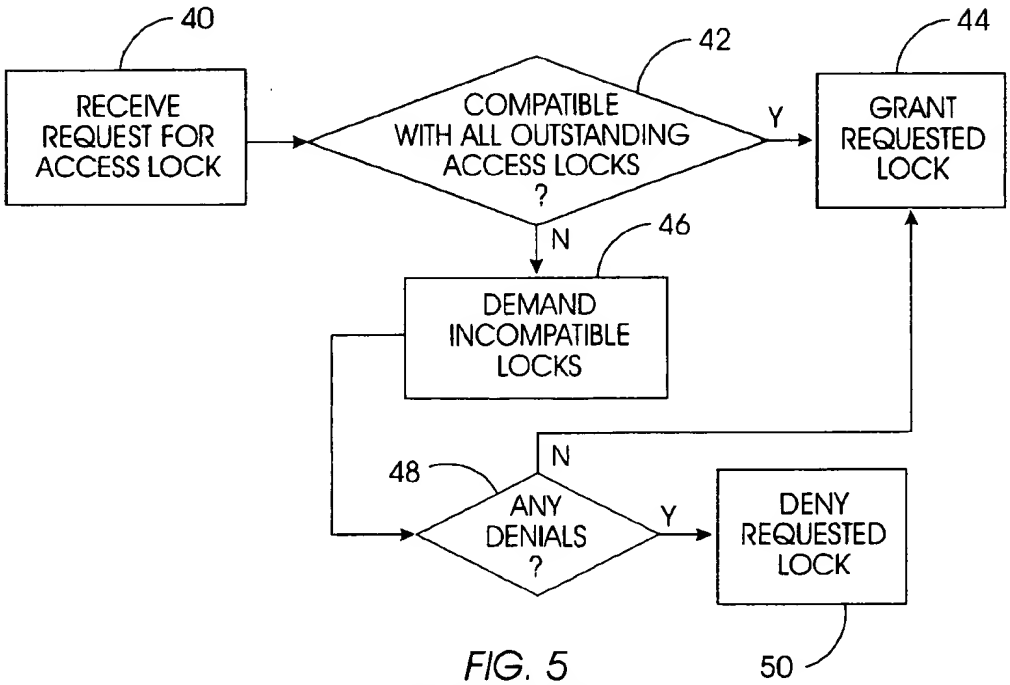


FIG. 5
SERVER LOGIC

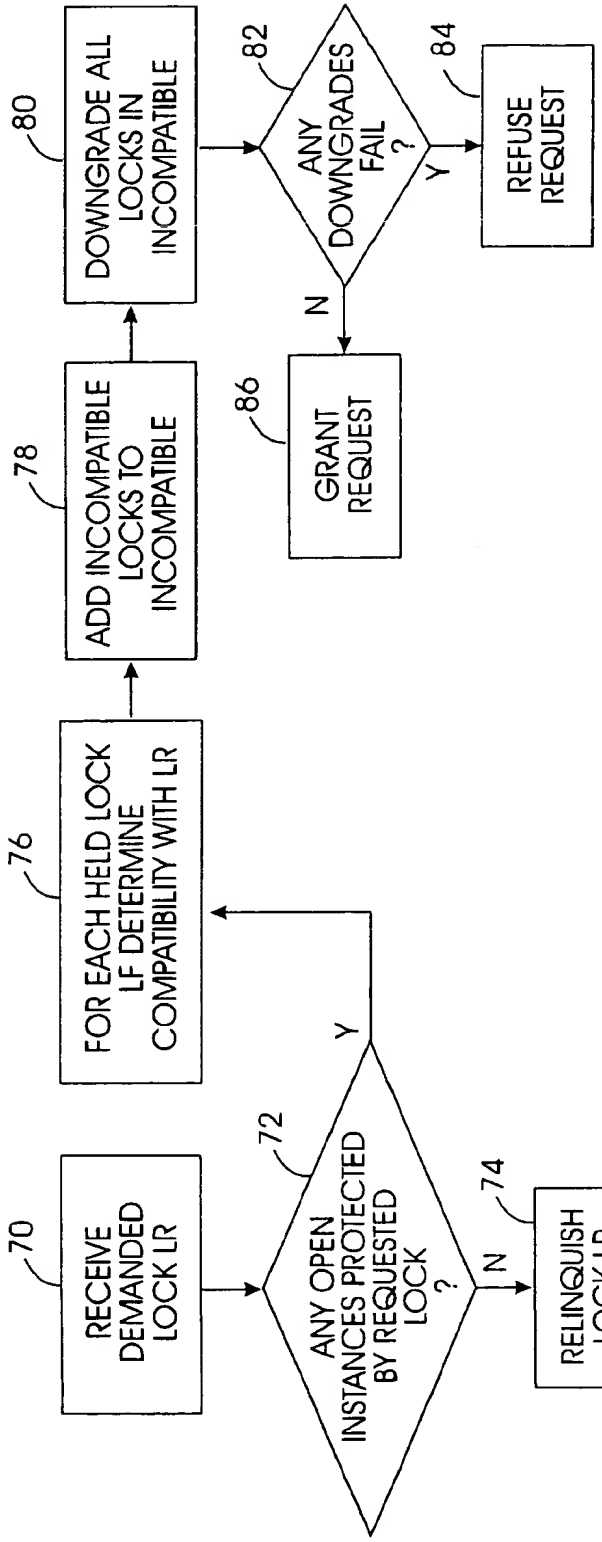


FIG. 6
CLIENT LOGIC FOR PROCESSING SEMI-PREEMPTIBLE LOCK DEMAND

SYSTEM FOR MANAGING ASSET ACCESS IN A DISTRIBUTED STORAGE SYSTEM

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to managing access to assets in distributed data storage systems such as file systems and databases.

2. Description of the Related Art

Distributed file systems are used to provide data sharing in distributed computer systems. Such systems centralize data storage, which improves the scalability and manageability of data access control. Moreover, centralized data storage also facilitates, among other things, easier storage device replacement and data backups, as compared to systems in which data storage is fragmented among local storage devices of many computers. It is to be understood that while, for disclosure purposes, the present discussion focuses on file systems, the principles set forth herein apply equally to other distributed data storage systems, such as distributed database systems.

To synchronize data access such that users share consistent views of shared data, requests from users to read and write data typically are sent to a central file server. The file server then manages access to the data using "locks" to ensure, e.g., that one user is not updating shared data by writing to it while another user might read an out-of-date version of the same data. In distributed systems, locks ordinarily are preemptible, in that the server can demand a lock previously provided by the server to one user to enable the user to access a file, and then give the lock to a second user. Unfortunately, requiring a central server to actively manage all data access degrades performance as compared to accessing a local data cache, since the server essentially represents a bottleneck.

As recognized by the present invention, while a central server can be used to manage synchronized access and coherent views of data, to optimize system performance the server should not be used as a target for all read and write requests. Stated differently, the present invention recognizes that it is desirable to provide local storage system semantics in a distributed environment, wherein communications with a server is minimized where possible. In this way, the speed, ease, and efficiency of accessing assets in a distributed storage network can approach that of accessing data in a local cache. Furthermore, the present invention understands that any asset locking scheme preferably be amenable to simplification, to further improve system performance.

SUMMARY OF THE INVENTION

A general purpose computer is programmed according to the inventive steps herein to manage access to assets in a distributed storage system. The invention can also be embodied as an article of manufacture—a machine component—that is used by a digital processing apparatus and which tangibly embodies a program of instructions that are executable by the digital processing apparatus to execute the present logic. This invention is realized in a critical machine component that causes a digital processing apparatus to perform the inventive method steps herein.

The invention can be implemented by a computer system including at least one general purpose client computer, at least one general purpose server computer, and a distributed data storage system accessible to at least the client computer.

The system also includes logic that can be executed by the client computer for undertaking method acts to manage access to assets in the storage system. The method acts undertaken by the client computer include sending a request for a first semi-preemptible access lock from the client computer to the server computer. The access lock can be thought of as a distributed lock that encapsulates local open instances protected by non-preemptible local locks, also referred to as file locks.

Also, the method acts include receiving the first access lock from the server computer, it being understood that the access lock pertains to at least one asset in the storage system. The asset is characterized by either an open state or a closed state. A demand can be subsequently received from the server computer for the first access lock, and the method includes selectively not relinquishing the first access lock if the open state exists for the asset, and otherwise relinquishing the first access lock.

The preferred method undertaken by the client computer includes, if an open state exists for the asset, attempting to downgrade incompatible locks held by the client computer, with the incompatible locks being characterized as being incompatible with the first access lock. The first access lock is not relinquished if any incompatible lock cannot be downgraded.

In another aspect, a computer system includes at least one general purpose client computer, at least one general purpose server computer, and a distributed data storage system accessible to at least the client computer. The system also includes logic that can be executed by the server computer for undertaking method acts to manage access to assets in the storage system. The method acts undertaken by the server computer include receiving a request for a first semi-preemptible access lock from a first client computer, and determining at least whether the first lock is compatible with a second semi-preemptible lock associated with a second client computer. Also, the logic includes granting the request if the first lock is compatible with the second lock, and otherwise demanding the second lock.

In still another aspect, a computer-implemented method for managing access among plural client computers to assets in a distributed data storage system associated with at least one server computer includes issuing semi-preemptible access locks to client computers. In accordance with present principles, the semi-preemptible access locks are conditions precedent for the grant of a file lock to open a file. The semi-preemptible access locks are relinquished upon demand of the server computer when no associated file lock is invoked.

The details of the present invention, both as to its structure and operation, can best be understood in reference to the accompanying drawings, in which like reference numerals refer to like parts, and in which:

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a schematic diagram showing the system of the present invention;

FIG. **2** is a table showing lock semantics;

FIG. **3** is a table showing lock compatibilities for an exemplary locking scheme;

FIG. **4** is a schematic representation of the legal upgrades and downgrades between locks;

FIG. **5** is a flow chart showing the logic executed by the server computer; and

FIG. **6** is a flow chart showing the logic executed by the client computer for processing a demand from the server.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring initially to FIG. **1**, a system is shown, generally designated **10**, for managing data access in a distributed data storage system, such as a storage area network (SAN) having associated client computers and at least one server computer. As shown, the system **10** can include a cluster of server computers, and the network can include plural storage disks and tapes and other storage devices. One or more of the disks can be "local" to a client computer, i.e., the client computer manages one or more disks as though the disks were local to the client computer.

In one intended embodiment, the computers of the present invention may be personal computers made by International Business Machines Corporation (IBM) of Armonk, N.Y., or the computers may be any computer, including computers sold under trademarks such as AS400, with accompanying IBM Network Stations. Or, the computers may be Unix computers, or OS/2 servers, or Windows NT servers, or IBM workstations or IBM laptop computers.

The flow charts herein illustrate the structure of the logic executed by the computers of the present invention as embodied in computer program software. Those skilled in the art will appreciate that the flow charts illustrate the structures of logic elements, such as computer program code elements or electronic logic circuits, that function according to this invention. Manifestly, the invention is practiced in its essential embodiment by a machine component that renders the logic elements in a form that instructs a digital processing apparatus (that is, a computer) to perform a sequence of function steps corresponding to those shown.

In other words, the flow charts may be embodied in a computer program that is executed by a processor within the computers as a series of computer-executable instructions. These instructions may reside, for example, in a program storage device **12** of the computers. The program storage device **12** may be RAM of the computers, or a magnetic or optical disk or diskette, DASD array, magnetic tape, electronic read-only memory, or other appropriate data storage device. In an illustrative embodiment of the invention, the computer-executable instructions may be lines of compiled C^{++} compatible code.

To better understand the flow charts described below that illustrate the present invention, reference is first made to FIGS. **2–4**. As a preferred but non-limiting example of the types of semi-preemptible access locks that can be used in the present invention, attention is now directed to FIG. **2**, which shows a table **14** of locks and lock semantics. It is to be understood that a semi-preemptible access lock of the present invention permits predefined open accesses to assets in the data storage system as long as the semi-preemptible access lock is held by a client computer. That is, to access an asset a client computer first obtains a semi-preemptible access lock, and then, as further described below, the client computer can permit processes to obtain file locks as required to instantiate actual open instances. Once an actual open instance is closed and the file lock relinquished, the client computer nonetheless retains the semi-preemptible access lock to support subsequent open instances until such time as the semi-preemptible access lock is relinquished in accordance with the disclosure below.

As shown, six locks, respectively named "metadata", "read", "shared", "write", "update", and "exclusive" can be provided from which a client computer can select, depending on the type of access to an asset that is desired by the client computer and the types of other concurrent open instances of the asset the client computer is willing to accept.

Accordingly, as indicated in the third column of the table **14**, the "M" semi-preemptible access lock can be used to access metadata of an asset under the lock, and when the "M" lock is used another client computer concurrently can access the same asset for any other type of open instance, i.e., READ, METADATA, and WRITE. Furthermore, the "R" lock can be used to obtain read accesses of an asset, and when the "R" lock is used another client computer concurrently can access the same asset for any other type of open instance. On the other hand, when the "S" lock is used, read accesses of an asset can be obtained under the lock, and when the "S" lock is used another client computer can concurrently access the same asset but only for read accesses and metadata accesses.

As further shown in FIG. **2**, the "W" lock can be used to obtain both read and write accesses of an asset, with any other concurrent access of the asset by another client computer being permissible. Moreover, the "U" lock can be used to obtain read and write accesses of an asset, and when the "U" lock is used another client computer concurrently can access the same asset but only for read and metadata accesses. On the other hand, when the "X" lock is used, read and write accesses of an asset can be obtained under the lock, and when the "X" lock is used another client computer can concurrently access the same asset but only for metadata accesses. The set of access privileges granted by a lock "L" can be designated "P_L". In contrast, the set of sharing privileges restricted by a lock "L" can be designated "C_L".

FIG. **3** illustrates a compatibility table **16**, which shows which locks are compatible with which other locks. Check marks indicate compatibility. As intended by one preferred embodiment, two locks are compatible with each other if they mutually share the access modes that the other lock protects. Stated differently, in one presently preferred embodiment lock L_s is compatible with lock L_T iff $P_{Ls} \subseteq C_{Lt}$ and $P_{Lt} \subseteq C_{Ls}$.

Thus, for example, the "M" lock is compatible with all other locks that might happen to have been granted, the "R" lock is compatible with all other locks but the "X" lock, the "W" lock is compatible with the "M", "R", and "W" locks, the "S" lock is compatible with the "M", "R", and "S" locks, the "U" lock is compatible with the "M" and "U" locks, and the "X" lock is compatible only with other outstanding "X" locks.

As set forth further below, locks may require upgrading or downgrading. FIG. **4** shows the legal upgrades and downgrades between the MSRWUX locks. For example, as indicated by the arrows the "X" lock can be upgraded to any other lock, the "U" lock can be upgraded to any other lock but the "X" lock, the "W" and "S" locks can be upgraded to the "R" and "M" locks, and the "R" lock can be upgraded only to the "M" lock. In contrast, the "M" lock can be downgraded to any other lock, and the "R" lock can be downgraded to the "W", "S", and "U" locks.

FIG. **5** shows the server logic that is executed when a request for a semi-preemptible access lock L_R is received by the server. Commencing at block **40**, a request for an access lock is received. Moving to decision diamond **42**, the server determines whether the requested lock is compatible with any other outstanding access lock. If it is determined at decision diamond **42** that the requested lock is compatible with all outstanding access locks, the process moves to block **44** to grant the requested lock.

In contrast, if the test at decision diamond **42** is negative, the logic moves to block **46** to demand all incompatible locks from the client computers that hold those locks. If any

denials are received at decision diamond **48**, the requested lock is denied at block **50**; otherwise, the lock is granted at block **44**.

FIG. **6** shows the logic executed by a client computer when a demand for a semipreemptible lock is received from the server. Commencing at block **70**, the demand is received, and at decision diamond **72** it is determined whether any open instances exist that are protected by the demanded lock, i.e., whether any children nodes representing local locks exist under the root node representing the demanded lock in the client forest. If not, the lock is relinquished at block **74**.

On the other hand, if open instances exist that are protected by the demanded semi-preemptible access lock, the logic flows to block **76** to determine the compatibility of each semi-preemptible access lock held by the client computer vis-a-vis the demanded lock. Proceeding to block **78**, all locks that are incompatible with the demanded lock are added to an INCOMPATIBLE list, and then, at block **80**, each lock in the INCOMPATIBLE list is attempted to be downgraded in accordance with the downgrades shown in FIG. **4**, while still protecting any local instances, i.e., while still encapsulating any local file locks. If it is determined at decision diamond **82** that any downgrades failed, the requested lock is refused to be relinquished at block **84**; otherwise, if all incompatible locks can be successfully downgraded as described further below, the client computer relinquishes the requested lock at block **86**.

Should a client computer receive a request for a local open instance that requires a stronger access lock than the one held by the client computer, it invokes the logic above to request the required access lock. As recognized herein, the client never needs to upgrade from a held lock to a stronger incompatible lock, because that would mean the client is not using the full strength of its current access lock. Clients address this situation by downgrading their current access lock to an access lock that protects existing open instances, and then upgrading to the needed stronger lock.

While the particular SYSTEM FOR MANAGING ASSET ACCESS IN A DISTRIBUTED STORAGE SYSTEM as herein shown and described in detail is fully capable of attaining the above-described objects of the invention, it is to be understood that it is the presently preferred embodiment of the present invention and is thus representative of the subject matter which is broadly contemplated by the present invention, that the scope of the present invention fully encompasses other embodiments which may become obvious to those skilled in the art, and that the scope of the present invention is accordingly to be limited by nothing other than the appended claims, in which reference to an element in the singular means "at least one". All structural and functional equivalents to the elements of the above-described preferred embodiment that are known or later come to be known to those of ordinary skill in the art are expressly incorporated herein by reference and are intended to be encompassed by the present claims. Moreover, it is not necessary for a device or method to address each and every problem sought to be solved by the present invention, for it to be encompassed by the present claims. Furthermore, no element, component, or method step in the present disclosure is intended to be dedicated to the public regardless of whether the element, component, or method step is explicitly recited in the claims. No claim element herein is to be construed under the provisions of 35 U.S.C. §112, sixth paragraph, unless the element is expressly recited using the phrase "means for".

We claim:

1. A computer system, comprising:

at least one general purpose server computer;

at least one general purpose client computer;

a distributed data storage system accessible to at least the client computer; and

logic executable by the client computer for undertaking method acts to manage access to assets in the storage system, the method acts comprising:

sending a request for a first access lock from the client computer to the server computer;

receiving the first access lock from the server computer, the access lock pertaining to at least one asset in the storage system, the asset being characterized by either an open state or a closed state;

receiving a demand from the server computer for the first access lock; and

in response to the demand, selectively not relinquishing the first access lock if the open state exists for the asset, and otherwise relinquishing the first access lock.

2. The system of claim **1**, wherein the method acts undertaken by the logic further include:

if an open state exists for the asset, attempting to downgrade incompatible locks held by the client computer, the incompatible locks being characterized as being incompatible with the first access lock; and

not relinquishing the first access lock if any incompatible lock cannot be downgraded.

3. The system of claim **1**, wherein a lock is downgraded to a downgraded lock only when the downgraded lock protects open instances at the client computer.

4. A computer system, comprising:

at least one general purpose server computer;

at least first and second general purpose client computers;

a distributed data storage system accessible to at least the client computers; and

logic executable by the server computer for undertaking method acts to manage access to assets in the storage system, the method acts comprising:

receiving a request for a first access lock from the first client computer;

determining at least whether the first lock is compatible with a second lock associated with the second client computer;

granting the request if the first lock is compatible with the second lock; otherwise

demanding the second lock.

5. A computer program device comprising:

a computer program storage device readable by a client computer; and

a program on the program storage device and including instructions executable by the client computer for managing access to assets in a distributed data storage system, the program comprising:

computer readable code means for sending a request for a first semi-preemptible access lock from the client computer to a server computer;

computer readable code means for receiving the first access lock from the server computer, the access lock pertaining to at least one asset in the storage system, the asset being characterized by either an open state or a closed state;

computer readable code means for receiving a demand from the server computer for the first access lock; and

computer readable code means for selectively not relinquishing the first access lock if the open state exists for the asset, and otherwise relinquishing the first access lock.

6. The device of claim 5, wherein the program further comprises:

computer readable code means for, if an open state exists for the asset, attempting to downgrade incompatible locks held by the client computer, the incompatible locks being characterized as being incompatible with the first access lock; and

computer readable code means for not relinquishing the first access lock if any incompatible lock cannot be downgraded.

7. The device of claim 6, wherein a lock is downgraded to a downgraded lock only when the downgraded lock protects open instances at the client computer.

8. A computer program device comprising:

a computer program storage device readable by a server computer; and

a program on the program storage device and including instructions executable by the server computer for managing access to assets in a distributed data storage system, the program comprising:

computer readable code means for receiving a request for a first semi-preemptible access lock from a first client computer;

computer readable code means for determining at least whether the first lock is compatible with a second semi-preemptible lock associated with a second client computer;

computer readable code means for granting the request if the first lock is compatible with the second lock; otherwise demanding the second lock.

9. A computer-implemented method for managing access among plural client computers to assets in a distributed data storage system associated with at least one server computer, comprising the acts of:

issuing semi-preemptible access locks to client computers, the semi-preemptible access locks being conditions precedent for the grant of a file lock to open a file, the semi-preemptible access locks being relinquished upon demand of the server computer when no associated file lock is invoked.

10. The method of claim 9, further comprising the act of:

determining at least whether a requested semi-preemptible lock is compatible with an outstanding semi-preemptible lock, and if so, granting the requested lock without demanding the outstanding lock, and otherwise demanding the outstanding lock.

* * * * *

US 20020065829A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2002/0065829 A1

Neuhaus et al. (43) Pub. Date: May 30, 2002

(54) APPARATUS AND METHOD FOR SYNCHRONIZING DATABASES IN DISTRIBUTED COMMUNICATION SYSTEMS

(76) Inventors: Ralf Neuhaus, Lunen (DE); Rainer Uecker, Mulheim (DE)

Correspondence Address:
BELL, BOYD & LLOYD, LLC
P. O. BOX 1135
CHICAGO, IL 60690-1135 (US)

(21) Appl. No.: 09/954,833

(22) Filed: Sep. 17, 2001

(30) Foreign Application Priority Data

Sep. 19, 2000 (DE)..... 10046319.3

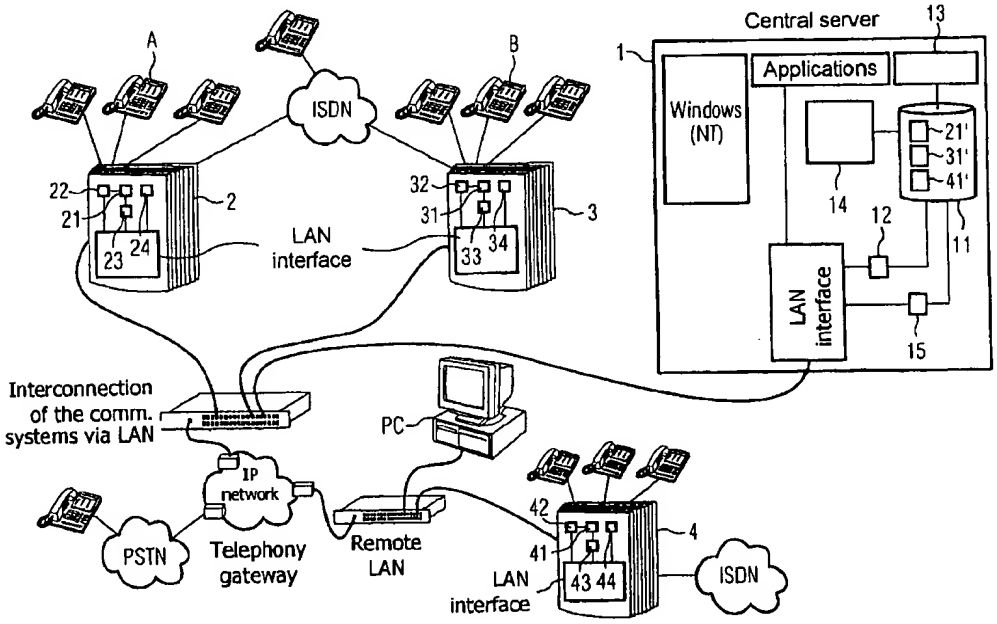
Publication Classification

(51) Int. Cl.[7] G06F 7/00

(52) U.S. Cl. 707/100

(57) ABSTRACT

Apparatus and method for synchronizing databases in distributed communication systems utilizing a server for a preferably private communication system having a number of communications installations connected to one another via a network, the server including a central database for centrally storing data for the individual communications installations and a central synchronization device which synchronizes the data between the central database and the individual communications installations.



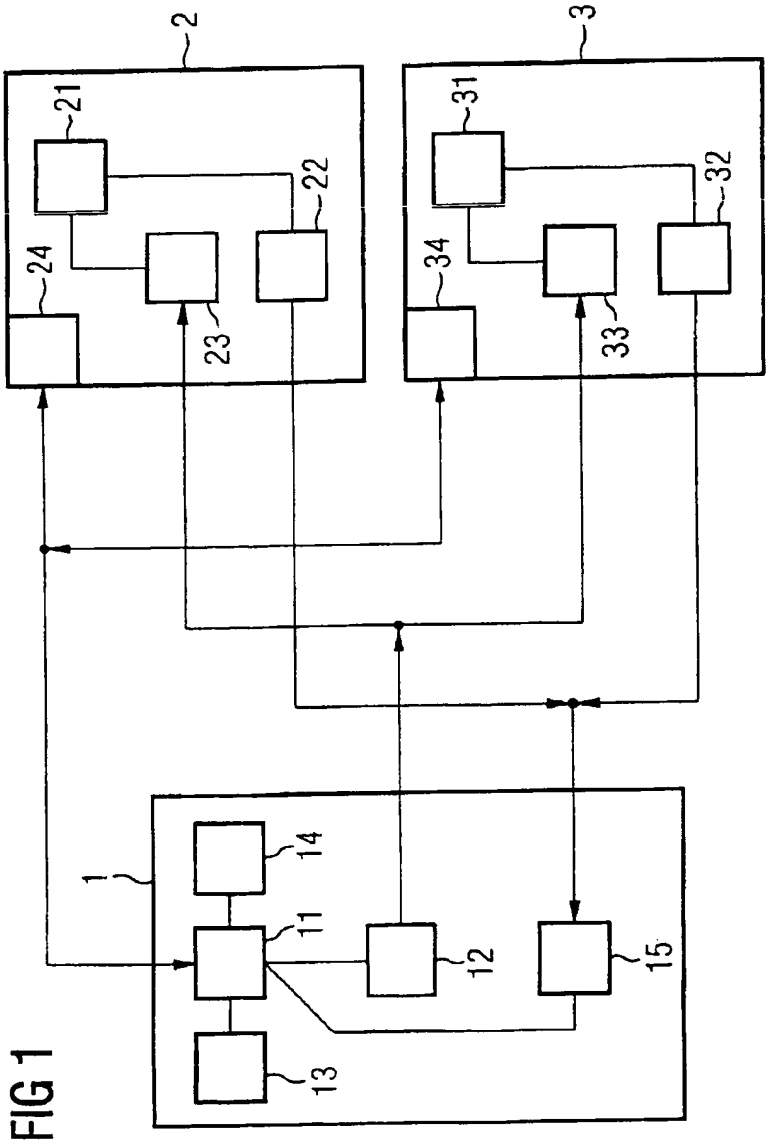


FIG 1

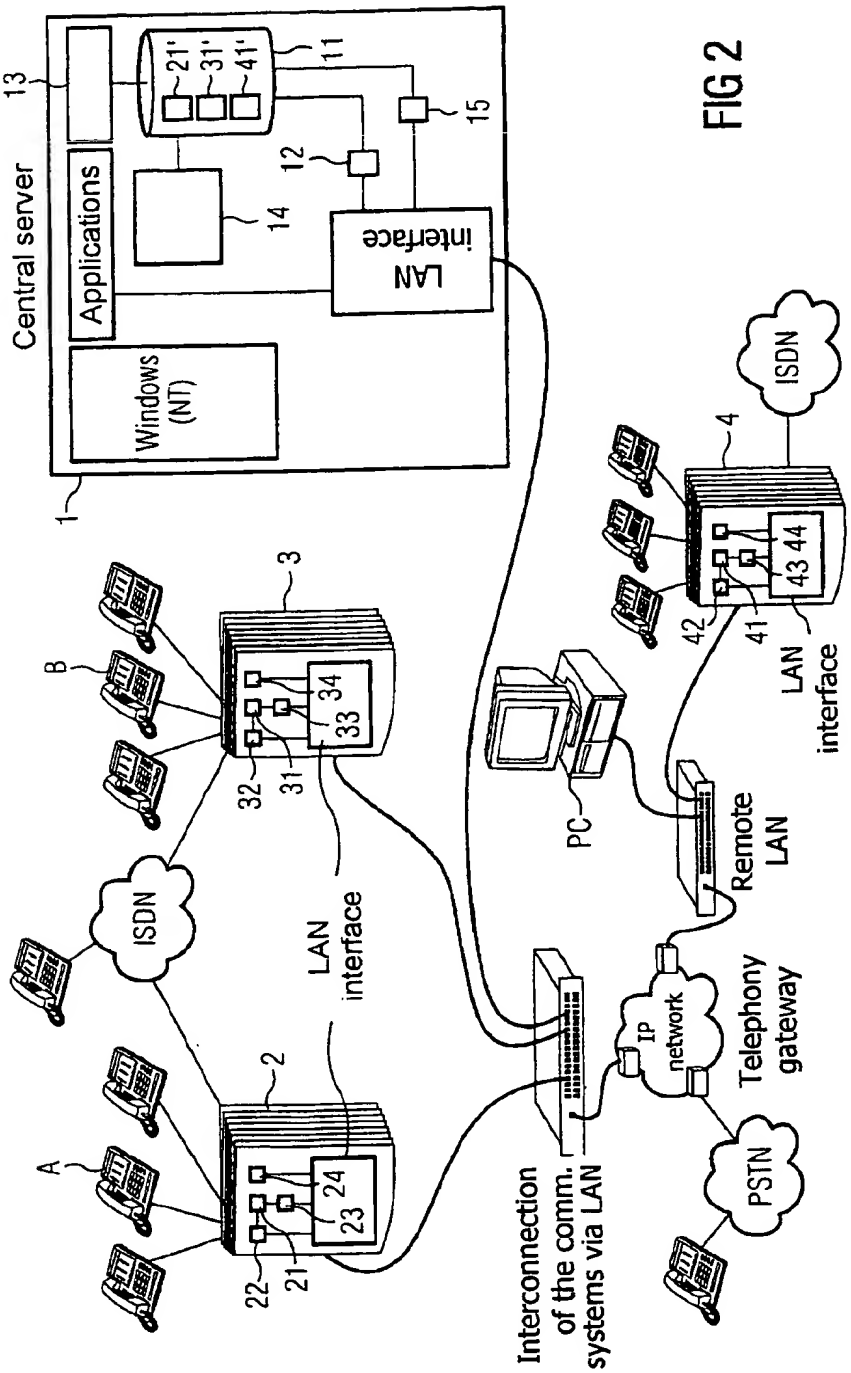

Central server
Applications
Windows (NT)
LAN interface
13
21'
31'
41'
11
15
12
14
1
3
B
32
31
33
34
ISDN
LAN interface
2
A
22
21
23
24
Interconnection of the comm. systems via LAN
PC
Remote LAN
IP network
Telephony gateway
PSTN
4
ISDN
42
41
43 44
LAN interface

FIG 2

APPARATUS AND METHOD FOR SYNCHRONIZING DATABASES IN DISTRIBUTED COMMUNICATION SYSTEMS

BACKGROUND OF THE INVENTION

[0001] The present invention relates to a server for a communication system including one or more interconnected communications installations, a communications installation for this communication system and a method for synchronizing databases arranged in the communications installations and/or the server.

[0002] Modem communication systems frequently include a number of physically distributed, interlinked communications installations. In this context, each of these communications installations has its own switching intelligence and its own local database for storing data specific to the communications installation. The local databases specific to the communications installations are in this case used for storing data required for independent operation (i.e., standalone operation) of a communications installation. For example, this data may include subscriber numbers, authorizations, exchange lines, direct-dial numbers, call numbers, configuration data and shortcode dialing destinations. To produce a network interconnection of networked communications installations from these communications installations, the individual communications installations each need to be configured appropriately. That is, they need to be brought into line with the data stored in the other communications installations.

[0003] To achieve the appropriate configuration, it is necessary, in particular, to allocate a consistent and unique call number plan for the entire communication system between the individual communications installations. Other data that is valid across the network interconnection also needs to be configured appropriately and synchronized in each of the individual communications installations.

[0004] One disadvantage of this configuration is that maintaining and servicing the local databases is very complex and is prone to error by virtue of the quantity of data which needs to be managed. For example, changing the call number plan in one of the communications installations requires that these changes also be reflected in the other communications installations of the communication system. In addition, manual administration of the communications installations can result in a call number being allocated more than once within the communication system.

[0005] An object of the present invention is, therefore, to provide a method and a device which permit simple administration and synchronization of databases within a communication system.

SUMMARY OF THE INVENTION

[0006] On the basis of the present invention, a central server includes a central, all-embracing server database storing at least some of the data stored in local client databases of the respective communications installations. The central database thus contains a depiction of at least some of the data of the respective local databases. The data specific to the communications installations (first data) includes information necessary for operating the respective communications installation.

[0007] One advantage of the present invention is that a preferably private communication system having interlinked communications installations looks like a single communication system from the outside. This allows central management and administration of the local databases of the individual communications installations to be conducted in a simple manner.

[0008] Advantageously, the server includes an administration device allowing central management and administration of the central and local databases. In this context, first data changed in the central database is transmitted to the appropriate communications installation, where the first data of the local database is updated.

[0009] The first data changed in the central database may be data which affects a number of communications installations within the communication system. Therefore, the server of the present invention contains a central checking device which checks whether changed first data are data affecting a number of communications installations and which updates these data for the appropriate communications installations in the central database if the result of the check is positive. The changed data are then synchronized with the respective local databases by a central synchronization device, (i.e., the data is sent to the respective communications installations, in which the data in the local databases is updated).

[0010] Changes to call number plans are examples of first data which, when changed in a local database of a communications installation, entail a change in further local databases of further communications installations. A call number plan contains information about which call numbers of the communication system are associated with which communications installation. In the event of a change in the call number plan of a communications installation, the new call number plan needs to be appropriately adjusted in all other communications installations within the communication system. Otherwise, unique connection setup (i.e., the unique assignment of a call number to a communication device) is no longer possible if identical call numbers are allocated within the communication system.

[0011] The server and the communications installation of the present invention respectively have a central and local updating device for receiving the changed first data which has been sent by the respective other apparatus via the network. In this context, the central and local updating devices update the received data in the corresponding central or local database.

[0012] Optionally, the central database of the server can provide second data, which are not stored in the respective communications installations and are not directly needed for operating a communications installation. A local access device provides the communications installation of the present invention with access to these second data. The second data can likewise be centrally administered and managed using the central administration device in the server.

[0013] Additional features and advantages of the present invention are described in, and will be apparent from, the following Detailed Description of the Invention and the Figures.

BRIEF DESCRIPTION OF THE FIGURES

[0014] FIG. 1 shows a schematic illustration of a server and of communications installations in accordance with the principles of the present invention.

[0015] FIG. 2 shows an exemplary embodiment of a private communication system having a number of interlinked communications installations in accordance with the principles of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0016] FIG. 1 shows a schematic illustration of a server 1 in a preferably private communication system or communication network in accordance with the principles of the present invention. The server 1 is connected to one or more communications installations 2, 3 via a network. The communications installations 2, 3 each have a local database 21, 31 for storing data specific to the communications installations (hereinafter referred to as first data). In this context, the connecting lines with arrows indicate the direction in which messages or data are interchanged between the communications installations 2, 3 themselves or between the server 1 and the communications installations 2, 3. In the present exemplary embodiment, the network is an IP-oriented (Internet Protocol) network, in which data is transmitted using a transmission protocol based on the IP protocol.

[0017] The central unit of the server 1 is formed by a central database 11, which stores a depiction (copy) of the individual local databases 21, 31 of the respective communications installations 2, 3 or parts of these local databases 21, 31.

[0018] A central synchronization device 12 of the server 1 monitors the central database 11 for any change to first data, required for operating a communications installation 2, 3, for example, stored in the central database 11. If first data affecting one or more communications installations 2, 3 has been changed in the central database 11, then the changed first data is transmitted by the central sychronization device 12 to the appropriate communications installation(s) 2, 3 via the network. The communications installations 2, 3 receive the changed data via a local updating device 23, 33 which is arranged in the respective communications installation 2, 3 and is used to update the first data stored in the respective local database 21, 31.

[0019] The first data can be centrally administered and managed using a central administration device 13. A central checking device 14 of the server 1 checks whether changed first data affecting a communications installation 2, 3 result in a change to first data in other communications installations 2, 3 (e.g., when a call number plan is changed) and automatically updates these first data in the central database 11. The first data changed in this manner are then transmitted to the appropriate other communications installations 2, 3 via the network, as a result of which the first data stored in the local databases 21, 31 of the appropriate communications installations 2, 3 are updated.

[0020] If data in a local database 21, 31 in one of the communications installations 2, 3 is changed (e.g., an entry for a shortcode dialing destination), then this entry also needs to be updated appropriately in the central database 11. For this purpose, the communications installations 2, 3 contain a respective local synchronization device 22, 32 monitoring the appropriate local database 21, 31 for changes. In cases in which first data in a local database 21, 31 are changed, the local synchronization device 22, 32 transmits the changed first data to the server 1 via the network. The changed first data is received by a central updating device 15 and is entered into the central database 11.

[0021] In addition, the communications installations 2, 3 can use a respective local access device 24, 34 to access second data, stored in a central database 11, which is not available in the respective local databases 21, 31.

[0022] FIG. 2 shows one preferred embodiment of a distributed communication system having three communications installations 2, 3, 4 and a central server 1. Connected to the communications installations 2, 3, 4 are a respective number of communication terminals (e.g., telephones). In the present exemplary embodiment, the communications installations 2, 3, 4 are networked via a local area network (LAN) in which data are transmitted using the IP protocol. Alternatively, the communications installations 2, 3, 4 can also be networked using a tunneling mechanism via circuit-switched networks (e.g., an ISDN-oriented communication network). In this case, the messages of a networking protocol specific to the communications installation are, by way of example, packaged into appropriate messages of the transmission protocol (e.g., IP, ISDN) and are sent via the network.

[0023] In the present exemplary embodiment, the communications installations 2, 3, 4 are connected to one another via an Ethernet LAN and can interchange signaling and voice data via this Ethernet LAN. The individual communications installations 2, 3, 4 are connected to the LAN via a respective LAN interface implemented in each of the communications installations 2, 3, 4. Each communications installation 2, 3, 4 has a local database 21, 31, 41 for storing first data specific to the communications ion.

[0024] Incorporated in the central server 1 is an all-embracing central database 11 containing a depiction of all the first data 21', 31', 41' or of at least some of the first data of all the communications installations 2, 3, 4 arranged in the communications system. Between the server 1 and the communications installations 2, 3, 4, there is a network connection for bidirectional data interchange. A central synchronization device 12 of the server 1 and local synchronization devices 22, 32, 42 of the communications installations 2, 3, 4 ensure that, in the event of a change to first data in the central database 11 or in one of the local databases 21, 31, 41, the corresponding first data in the respective communications installations 2, 3, 4 or in the server 1 are automatically updated. The central database 11, combined in this way, represents a depiction of the whole communication system, formed from a number of communications installations 2, 3, 4, in the representation of a single system (Single System Image).

[0025] The central database 11 can be viewed, altered and/or configured by an appropriately designed central administration device 13. In addition, a central checking device 14 of the server 1 is used to provide a consistency check, so that, by way of example, it is not possible to allocate call numbers twice within the communications system. Besides the first data of the individual communica-

tions installations **2, 3, 4**, the central database **11** also contains additional information (hereinafter referred to as second data) relating to the communications installations **2, 3, 4**. The second data contains information such as the address of a communications installation **2, 3, 4** in the network, this additional information being significant to the communication network.

[0026] From the outside and from the administrative point of view, this possibly worldwide system interconnection acts like a single installation and can also be administered like such an installation. The central database **11** is likewise accessed for cross operations between the communications installations **2, 3, 4**. By way of example, when a first subscriber A on the communications installation **2** calls a second subscriber B on the communications installation **3**, the server **1** can provide all the necessary information (second data), such as name and address, relating to the desired second subscriber B. If the communications installation **2** does not know the location of the desired second subscriber B, then this can be requested via the central database **11** using the local access device **24, 34, 44**. In this way, it is not absolutely necessary for each communications installation **2, 3, 4** to have available all the locations of the subscribers in the whole communication system. Similarly, **FIG. 2** shows the central updating device **15**, explained with reference to **FIG. 1**, and the respective local updating devices **23, 33, 43**.

[0027] Joint central administration can also be carried out and used for individual systems which are at a great physical distance from one another (e.g., in other towns and countries) and are, for example, connected to one another via an IP network (Internet).

[0028] The server **1** can also provide additional applications, for example, to extend the functionality of the communications installations **2, 3, 4**, such as service features and CTI applications (Computer Telephony Integration). To this end, the server **1** holds an operating system (e.g., Windows NT) holding the individual software components.

[0029] Although the present invention has been described with reference to specific embodiments, those of skill in the art will recognize that changes may be made thereto without departing from the spirit and scope of the invention as set forth in the hereafter appended claims.

1. A server for a communication system, the server comprising:

a plurality of communications installations connected to one another via a network, the communications installations each having a local database for storing first data;

a central database for storing at least some of the first data stored in the local databases of the respective communications installations; and

a central synchronization device for monitoring the central database for changes to the first data affecting the communications installations and for transmitting the changed first data to the respective communications installation via the network.

2. A server for a communication system as claimed in claim 1, the server further comprising a central checking device for checking whether the changed first data affect a plurality of communications installations, and for updating the changed first data in the respective communications installations if a result of the check is positive.

3. A server for a communication system as claimed in claim 2, wherein the first data are call number plans affecting operation of a plurality of communications installations.

4. A server for a communication system as claimed in claim 1, the server further comprising a central updating device for receiving the first data which have been changed in a local database of a communications installation and have been transmitted to the server via the network, and for updating the first data in the central database.

5. A server for a communication system as claimed in claim 1, wherein the central database provides second data for the respective communications installations.

6. A server for a communication system as claimed in claim 5, the server further comprising a central administration device for centrally administering the first and second data in the central database.

7. A server for a communication system as claimed in claim 1, wherein the network is an IP-oriented network.

8. A communications installation for a communication system, wherein the communication system includes a plurality of communications installations connected to one another via a network, the communications installation comprising a local database for storing first data, and a local synchronization device for monitoring at least some of the data stored in the local database for changes and for transmitting changed first data via the network to a server arranged in the network.

9. A communications installation for a communication system as claimed in claim 8, the communications installation further comprising a local updating device for receiving the first data which affect the communications installation and have been changed in a central database of the server, and for updating the changed first data in the local database.

10. A communications installation for a communication system as claimed in claim 9, the communications installation further comprising a local access device for accessing second data stored in the central database.

11. A method for synchronizing both local databases arranged in communications installations and a central database of a server in a communication system, the communications installations and the server being connected to one another via a network, the method comprising the steps of:

storing a copy of at least some of the first data stored in the local databases of the respective communications installations in the central database;

monitoring the central database for changes to the first data affecting a communications installation;

transmitting changed first data to the respective communications installation via the network if which a change has been made to the first data;

monitoring the local data bases for changes to the first data; and

transmitting the changed first data from the appropriate communications installation to the server via the network if a change has been made to the first data.

12. A method for synchronizing databases as claimed in claim 11, the method further comprising the steps of:

receiving, at a communications installation, first data changed in the central database; and

updating the changed first data in the local database of the communications installation.

13. A method for synchronizing databases as claimed in claim 11, the method further comprising the steps of:

receiving, at the server, first data changed in a local database; and

updating the changed first data in the central database of the server.

14. A method for synchronizing databases as claimed in claim 11, the method further comprising the steps of:

checking the changed first data to determine whether the changed first data affect a plurality of communications installations; and

updating the first data in the appropriate communications installations if a result of the check is positive.

15. A method for synchronizing databases as claimed in claim 14, wherein the first data affecting a plurality of communications installations are call number plans.

16. A method for synchronizing databases as claimed in claim 11, the method further comprising the step of providing second data for the communications installations via the central database.

17. A method for synchronizing databases as claimed in claim 16, the method further comprising the step of accessing the second data stored in the central database via the communications installations.

* * * * *

US 20020082927A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2002/0082927 A1**

Borenstein et al. (43) **Pub. Date: Jun. 27, 2002**

(54) **INTELLIGENT CACHING ROUTERS**

(76) Inventors: **Nathaniel S. Borenstein**, Ann Arbor, MI (US); **Kurt Brown**, Ann Arbor, MI (US); **Keith Harder**, Ann Arbor, MI (US); **John Christopher Kaltwasser**, Ann Arbor, MI (US); **Greg Merriman**, Ann Arbor, MI (US); **Daniel Pritts**, Ann Arbor, MI (US)

Correspondence Address:
John G. Posa
Gifford, Krass, Groh et al
Suite 400
280 N. Old Woodward Ave.
Birmingham, MI 48009 (US)

(21) Appl. No.: **09/989,942**

(22) Filed: **Nov. 21, 2001**

Related U.S. Application Data

(63) Non-provisional of provisional application No. 60/252,848, filed on Nov. 22, 2000.

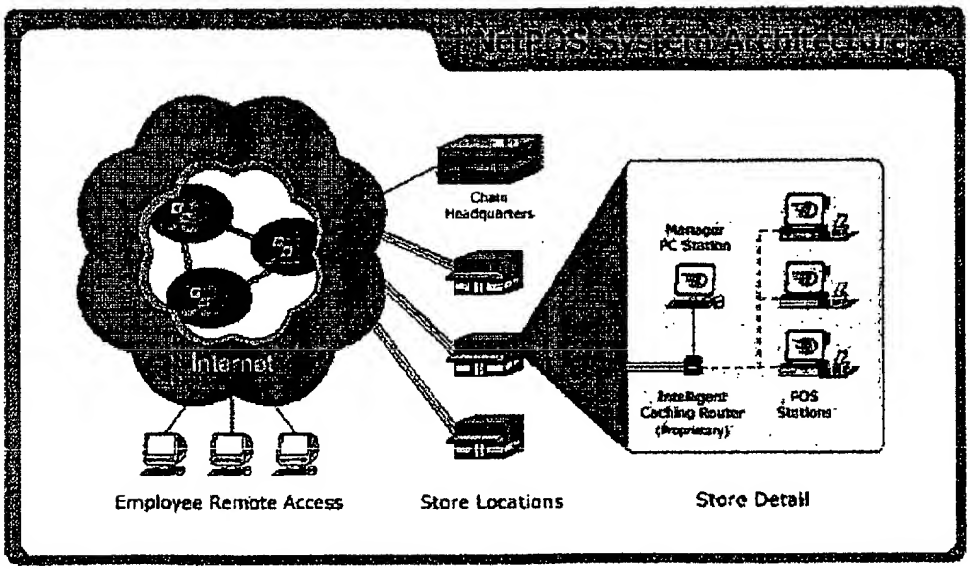
Publication Classification

(51) **Int. Cl.**[7] **G06F 17/60**; G06G 1/12; G06F 15/16

(52) **U.S. Cl.** **705/21**; 705/16; 709/227; 709/203

(57) **ABSTRACT**

An intelligent caching router (ICR) balances the cost-saving and functionality-enhancing benefits of the application-service-provider (ASP) model of software delivery against the inherent risks of relying on networked computing. In so doing, the ICR makes the ASP model practical for services that require extremely high levels of reliability and availability. The ICR is inserted functionally between a (thin) client and the network (i.e., Internet, intranet or extranet) and performs certain operations, including the logging of "mission-critical" application state data; network connectivity monitoring; traditional backup routing features; mission-critical server emulation; and server resynchronization upon reconnection. When networking problems are detected, the ICR initially takes steps to try and restore connectivity. In taking such actions, the ICR is largely behaving as a traditional intelligent network router. However, when such traditional backup routing fails, the ICR begins to act as a surrogate for the unreachable remote server on which the application service depends. In particular, for the application subset that the service providers have deemed "mission critical," the ICR makes application-specific responses to permit operations to continue, and logging the requests and response it has issued. When the communications link is restored, the ICR will re-synchronize with the remote server and then return to its normal "passive" operation. The invention is particularly suited to electronic commerce transactions, since accounting, crediting or debiting may be considered critical transactions, whereas other forms of updating, reporting, and the like are typically less critical. One disclosed example shows the role of an ICR in a point-of-sale application.



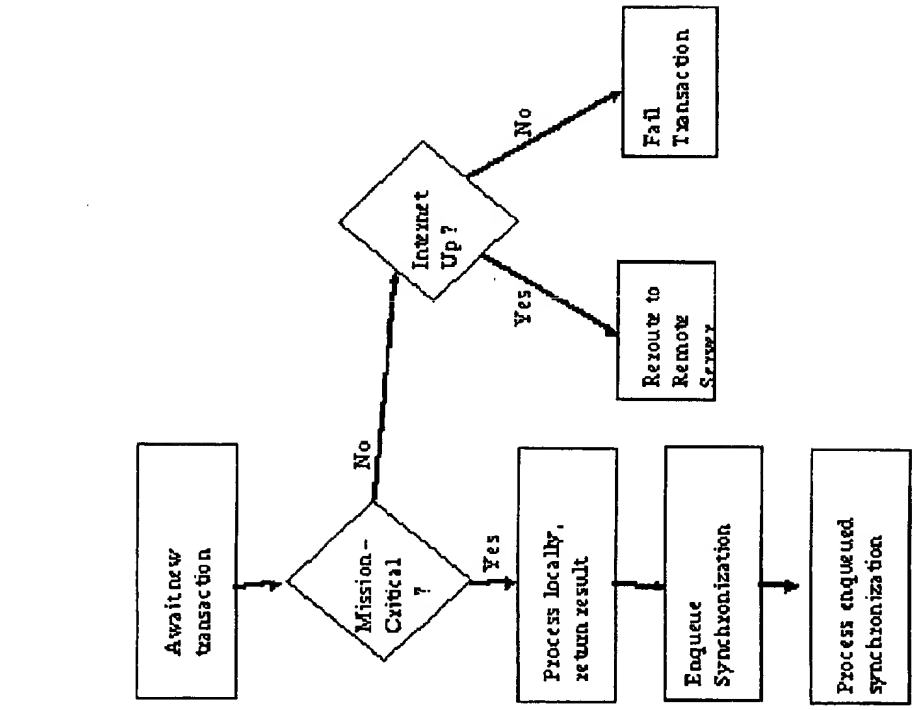


FIGURE 2

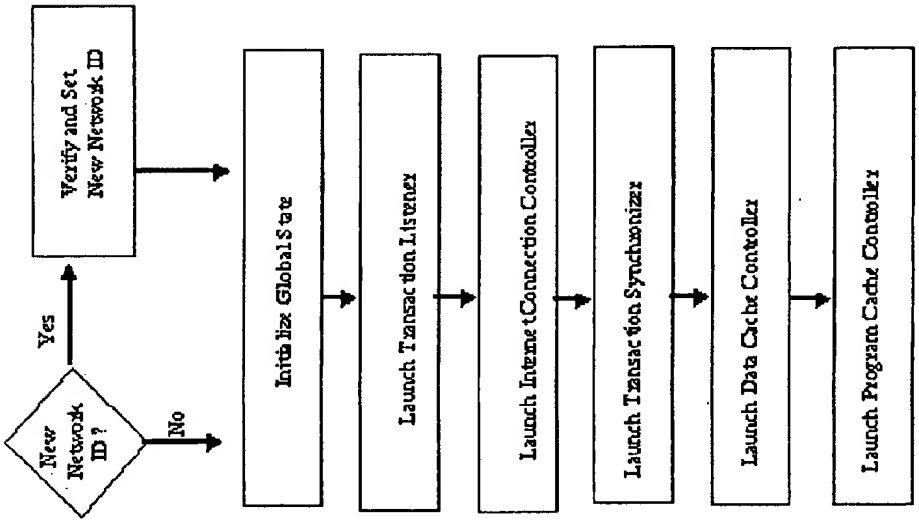


FIGURE 1

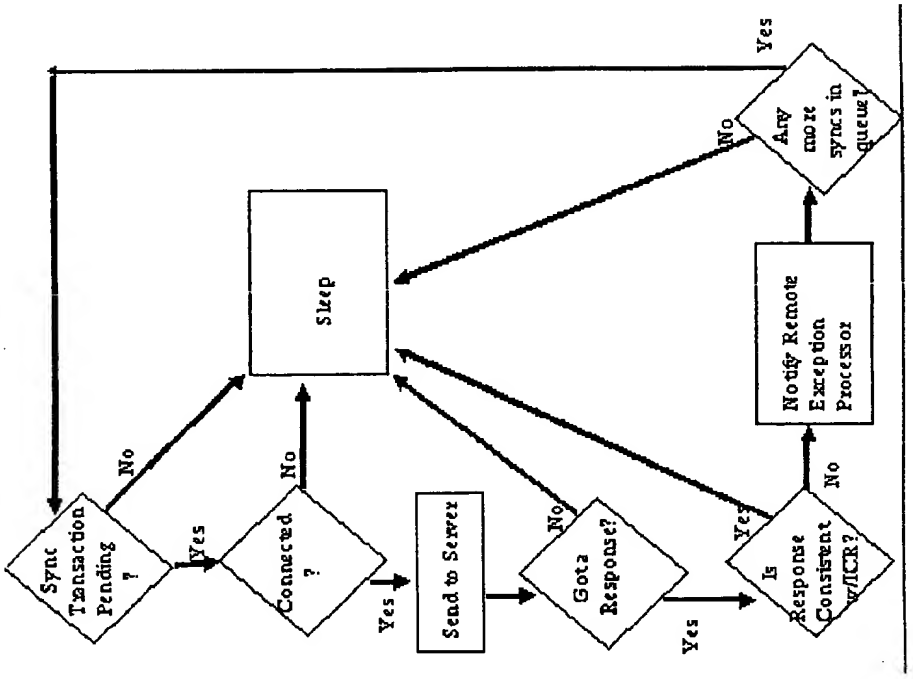


FIGURE 4

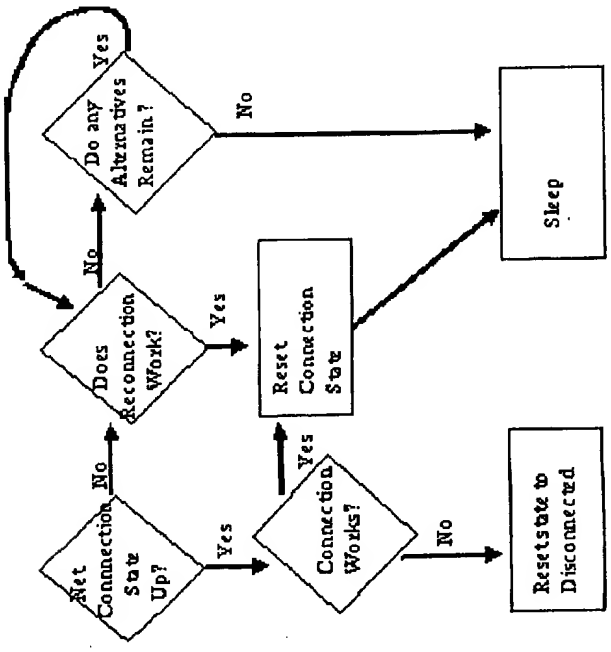


FIGURE 3

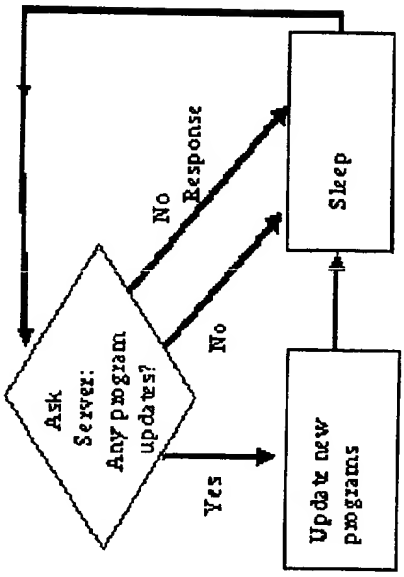


FIGURE 6

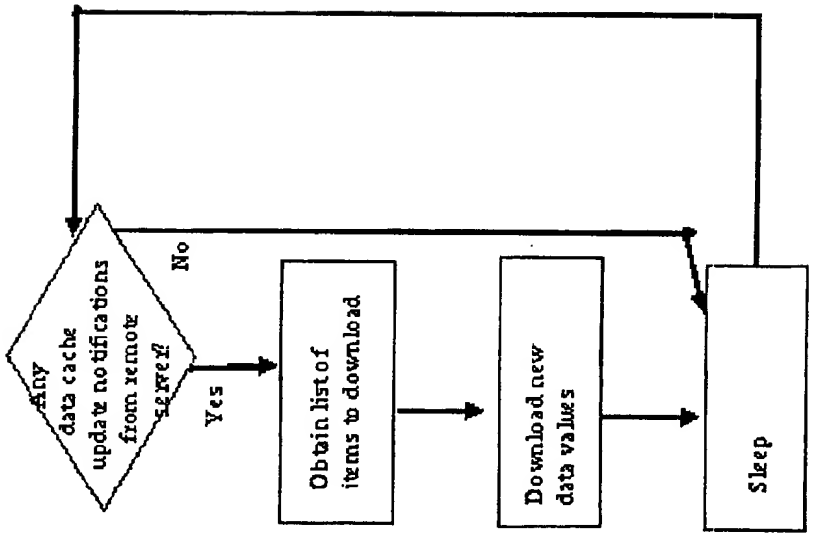


FIGURE 5

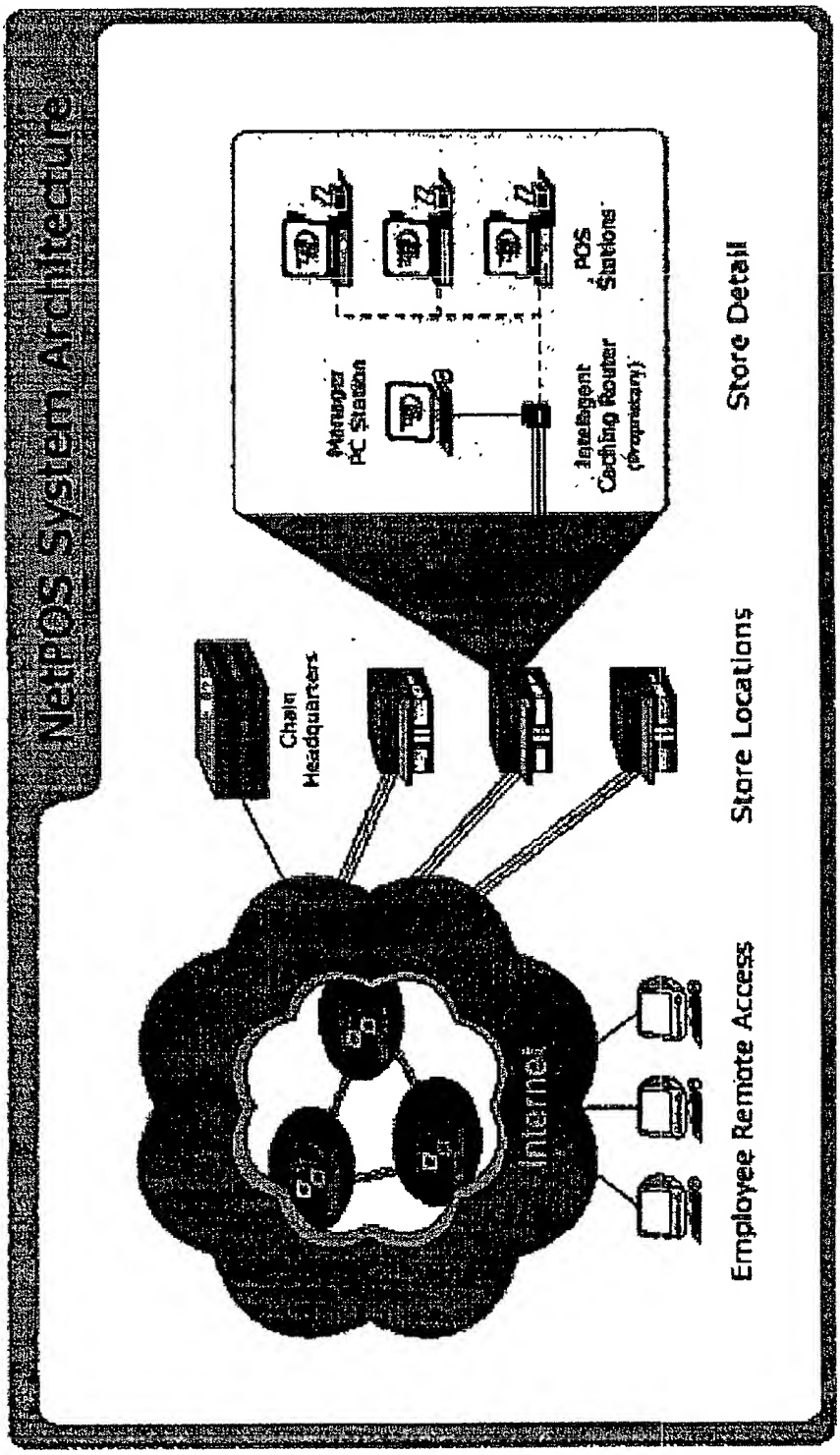

NetPOS System Architecture
Chain Headquarters
Internet
Employee Remote Access
Store Locations
Store Detail
Manager PC Station
Intelligent Caching Router (Proprietary)
POS Stations

FIGURE 7

INTELLIGENT CACHING ROUTERS

REFERENCE TO RELATED APPLICATION

[0001] This application claims priority from U.S. provisional application Serial No. 60/252,848, filed Nov. 22, 2000, the entire contents of which is incorporated herein.

FIELD OF THE INVENTION

[0002] The present invention relates generally to the maimer in which users at a physical site are given access to information services delivered over a network and, more particularly, to benefits associated with combining centralized services and administration while maintaining the high-reliabilty of locally-based services.

BACKGROUND OF THE INVENTION

[0003] The history of computing services shows a persistent and long-term tension between the advantages of localized and distributed computing. The introduction of time-sharing systems in the 1960's ushered in an era of shared access to centralized resources, in which relatively inexpensive terminal devices shared access to expensive centralized computing services. This model permitted enormous cost efficiencies and made computing power much more widely available.

[0004] The introduction of personal computers in the 1970's and 1980's demonstrated the benefits of the inverse approach: localized processing power was important for providing sophisticated user interfaces to increasingly complex applications, and was widely perceived as empowering users by reducing their dependence on a centralized computing enterprise and bureaucracy.

[0005] Both models persist because neither is clearly superior in all respects. Distributed computing is well known to create administrative and support nightmares, while centralized computing is well known to frustrate end users attempting to make creative new uses of computing resources, and to suffer dramatic, paralyzing losses of service in the event of network problems.

[0006] The advent of the World Wide Web has preserved this dichotomy while moving it onto a new environment, where standardized network protocols greatly increase the potential effective domain of any computing application. Accordingly, client/server applications using these protocols typically come in two models, locally-based and Application Service Provider (ASP). In a locally-based web application, the web server exists on-site, and Internet protocols are simply used as any other local network protocols might be used. In the ASP (Application Service Providers) model, the application server exists completely off-site, and is centrally administered and operated. Clearly a locally-based service is more reliable because it does not depend on wide-area networking, while an ASP-based service is far less work to operate and maintain at the local site.

[0007] Prior art in this area has focused almost entirely on unidirectional communication. One widely used technique is "web proxy caching," in which a locally sited server maintains a cache of information from a plurality of remote servers. When combined with mechanisms to ensure the cache's completeness, this mechanism can improve the reliability of a web-based application by providing disconnected access to all static data from the remote server. However, the unidirectional nature of a proxy cache does not meet the needs of transaction-based systems in which locally-originated transaction data needs to be processed and stored on a central server.

[0008] Another established technique is a "web mirror," in which an entire web site is maintained as a secondary copy of another, with relatively infrequent updates to preserve consistency. A web mirror has the advantage of always providing a complete copy of a remote database, but at the cost of being unable to guarantee its consistency with the central database, since updates are not automatically propogated. Moreover, like a proxy cache, a web mirror has no mechanism for ensuring the consistency of locally-originated transactions with the central database.

[0009] Earlier in the history of computing, similar problems were addressed through the use of "staging servers." A staging server is a computer server developed for the express purpose of performing intermediate local processing before application-level synchronization with a central server. Staging servers remain common in batch-oriented legacy applications, where they generally serve the purpose of consolidating and aggregating local transactions until the time arrives for such data to be uploaded to a central service. However, they are based on the expectation that such uploads are infrequent and that connectivity to the central service is sporadic and generally unavailable, which implies that a staging server must be treated as a full server in its own right for purposes of backup, administration, and system maintenance.

[0010] Fundamentally, however, existing systems require a major trade-off between the simplification and cost savings of an ASP-like model and the highest possible levels of application reliability and availability. This tradeoff made the ASP model unsuitable for any "mission-critical" software applications.

SUMMARY OF THE INVENTION

[0011] This invention improves upon the existing art by providing an intelligent caching router that is inserted functionally into the traditional ASP model, between the thin client and the network (i.e., Internet, intranet or extranet). Broadly, the ICR augments the existing routing technology to balance the cost-saving and functionality-enhancing benefits of the ASP model of software delivery against the inherent risks of relying on networked computing. In so doing, the ICR makes the ASP model practical for services that require extremely high levels of reliability and availability.

[0012] Generally speaking, each ICR implementation performs certain operations, including the logging of "mission-critical" application state data; network connectivity monitoring; traditional backup routing features; mission-critical server emulation; and server resynchronization upon reconnection. When networking problems are detected, the ICR initially takes steps to try and restore connectivity. In taking such actions, the ICR is largely behaving as a traditional intelligent network router. However, when such traditional backup routing fails, the ICR begins to act as a surrogate for the unreachable remote server on which the application service depends.

[0013] In particular, for the application subset that the service providers have deemed "mission critical," the ICR makes application-specific responses to permit operations to continue, and logging the requests and response it has issued. When the communications link is restored, the ICR will re-synchronize with the remote server and then return to its normal "passive" operation.

[0014] The invention is particularly suited to electronic commerce transactions, since accounting, crediting or debiting may be considered critical transactions, whereas other forms of updating, reporting, and the like are typically less critical. One disclosed example (of many), shows the role of an ICR in a point-of-sale application.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] **FIG. 1** is an outline of the initialization logic to be executed by an Intelligent Caching router (ICR) at startup;

[0016] **FIGS. 2 through 6** outline the logic of each of five process threads launched by the ICR at startup, wherein, in particular:

[0017] **FIG. 2** illustrates that when a transaction is received from the thin client, the ICR spawns a new thread, subroutine, procedure, or other process for responding to the request;

[0018] **FIG. 3** shows how the Internet Connection Control Process wakes up periodically and checks the globally-available settings describing the current state of the Internet connection;

[0019] **FIG. 4** shows how the Transaction Synchronization Process wakes up periodically and checks to see if there are any transactions are in the synchronization queue and if the Internet connection is functional;

[0020] **FIG. 5** shows how the Data Cache Control Process waits for the remote server to notify it of data that needs to be updated, and then downloads that data from the remote server;

[0021] **FIG. 6** shows how the Program Cache Control Process periodically polls the remote server to request a list of programs that needs to be updated, and then downloads those programs from the remote server; and

[0022] **FIG. 7** shows the role played by an ICR in the context of a larger application such as an Internet-based retail point-of-sale system.

DETAILED DESCRIPTION OF THE INVENTION

[0023] Before describing the invention in detail, the following terms will be introduced with respect to their roles in the new method:

DEFINITION OF TERMS

[0024] A "SOFTWARE APPLICATION" is any application of computing technology to provide a specific type of functionality to human users. It is distinguished from software that does not have, as its main purpose, direct satisfaction of a user-level request. Thus, for example, electronic mail, spreadsheets, and web browsing are considered software applications, while file systems, display drivers, and device or network management code are not.

[0025] A "REMOTE SERVER" is a computing engine that is located at a significant physical distance from the human user, substantially outside of the user's direct control. An "APPLICATION SERVICE PROVIDER (ASP)" is an entity that delivers software applications wide area delivery networks such as the Internet and its constituent networks, or similar networks. In the ASP model, most of the "intelligence", or software logic, takes place on a remote server, while the user's local machine (herein called the "THIN CLIENT") serves primarily to manage the human-computer interface (input and output).

[0026] An "INTELLIGENT CACHING ROUTER (ICR)" is a software or hardware object that interposes itself between the thin client and the remote server. It is itself a highly-specialized hybrid application, neither a pure client nor server, the only purpose of which is to increase the overall reliability of software applications delivered by ASP's over a sporadically unreliable remote connection.

[0027] "MISSION-CRITICAL" functionality is the subset of functionality, in a software application, that has the highest possible requirements for reliability and availability. The division of a software application into mission-critical and non-mission-critical subsets is highly application dependent, but has traditionally rarely been formalized.

ICR System Description

[0028] An ICR is a component that is inserted into the traditional ASP model, in between the thin client and the Internet (or Intranet/extranet):

THIN CLIENT←→ICR←→INTERNET←→REMOTE SERVER

[0029] Because it is co-located on the client end (possibly even as a software application running on the same hardware as the application), the link between the client and the ICR is fundamentally immune to communication difficulties inherent in the use of the Internet. In normal operation, an ICR functions as a passive observer of the traffic between the client and the server, logging only enough state to permit it to perform its primary function, the preservation of mission-critical functionality in the event of network connectivity problems, and monitoring the ongoing network traffic in order to quickly detect such problems when they occur.

[0030] When networking problems are detected, the ICR becomes more active. Initially, it will take steps to try to restore connectivity. Such steps may include, but are not limited to: Routing traffic to an alternative network provider; establishing communication via a backup link such as a dial-up line; or using paging or other independent communication technology to notify network administrators of an outage. In taking these actions, the ICR is largely behaving as a traditional intelligent network router. It is what the ICR does when traditional backup routing fails that differentiates an ICR from traditional routers. In such cases, an ICR begins to act as a surrogate for the unreachable remote server on which the application service depends.

[0031] For the application subset that the service providers have deemed "mission critical," the ICR makes application-specific responses to permit operations to continue, and logging the requests and response it has issued. When the communications link is restored, the ICR will re-synchronize with the remote server and then return to its normal "passive" operation.

[0032] The ICR augments hardware and software routing technology to include minimal emulation of application servers during transient communication outages. It provides a new way of balancing the cost-saving and functionality-enhancing benefits of the ASP model of software delivery against the inherent risks of relying on Internet-like wide area networking technology, and thus makes the ASP model practical for services that require extremely high levels of reliability and availability.

[0033] For example, one strong benefit of ASP-delivered software services is the elimination of the need for data backup at each remote site. In an ASP, the local machines—the thin clients—are interchangeable, so that a ruined disk may be easily replaced without concern for the data it contains. In a fully local service, on the other hand, local data backup is essential. The use of ICR actually involves only one potential point of data loss: if the hardware on which the ICR resides fails catastrophically during a period of network outages, some data will indeed be lost. However, even though this creates a small window for data loss in ASP services, such risk is in fact much smaller than the risk, in traditional software services, of the loss of all data newer than the last backup. Thus this risk is likely to be perceived as a very acceptable cost for the reduction of the risk of catastrophic connectivity loss. The risk can itself be further minimized by the ASP designer's choice of which data to consider so mission-critical as to make it locally-processed and hence subject to this kind of risk.

[0034] The introduction of an ICR into the ASP model for delivering information services also requires additional sophistication at the level of Internet connection management. Within the local site, the thin client machines will typically, in the preferred embodiment, be configured to address only the machines on the local network, one of which is the ICR. This configuration insulates the thin client machines from any dependence on the external network, but imposes on the ICR the necessity of managing the IP address space. An ICR will therefore often include the traditional functionality of a NAT (Network Address Translation) router. In general, the ICR is a logical place to include any routing or firewall functionality desired for the application, because it sits in the right spot architecturally as the only local machine that actually communicates to the Internet. It will also generally makes sense for the ICR to perform DNS (domain name system) lookup, DHCP (dynamic host configuration protocol) service, and any other network services that are essential for the functioning of the local thin client machines.

Basic Features

[0035] To implement an ICR for a particular software application, it is first necessary to determine which aspects of the application are to be considered mission-critical. This is essentially an additional design stage, in which the designer of the application service is empowered with relatively fine-grain control over the traditional tradeoff between simplicity and cost-effectiveness of local operations and the reliability and availability of the overall system service.

[0036] The invention is particularly suited to electronic commerce transactions, since accounting, crediting or debiting may be considered critical transactions, whereas other forms of updating, reporting, and the like are typically less critical. As one example, **FIG. 7** shows the role of an ICR in a point-of-sale application. Here, the ICR is used to permit sales transactions to continue to be processed in the event of Internet outages, while non-mission-critical functionality related to reporting, inventory data, or customer relationship management might be unavailable during the interruption.

[0037] According to the invention, each ICR implementation performs the following basic operations:

[0038] 1. Logging of "mission-critical" application state data. Every time a local user on the thin client terminal performs an operation that changes the state of the application, this information may optionally be responded to immediately by the ICR (to avoid user-level delays due to network problems) and then must be established as a permanent state change, by both changing the cached data in the ICR and ensuring that the authoritative data at the remote server is similarly changed. The latter action will be nearly immediate in the general case, but could be significantly delayed in the event of network problems. When such problems occur, the ICR is responsible for queuing up all such changes until connectivity is restored.

[0039] 2. Network monitoring and detection of outages. The ICR must pro-actively monitor network connectivity, to ensure that network problems are detected in a "background" mode rather than while a user is waiting for a mission-critical response.

[0040] 3. Traditional backup routing features. The ICR is responsible for choosing among a plurality of available mechanisms and routes by which to connect to the Internet, to establish backup connectivity whenever the primary or currently-used connectivity mechanism fails.

[0041] 4. Mission-critical server emulation. The ICR is able to act in place of the remote server during transient Internet outages, which means that it must maintain a cached copy of all mission-critical server data as well as a cached copy of the actual server processing code.

[0042] 5. Server resynchronization upon reconnection. After an Internet outage, the ICR must be able to resynchronize its state with the remote server, ensuring that all mission-critical processing that took place during the outage is properly reflected in the remote server's state information (typically its database).

System Security

[0043] In general, an ASP using an ICR has roughly the same security profile as any other ASP. For example, data must be encrypted for transport over the Internet if it is to be protected from the view of third parties, and the local client must authenticate itself to the remote server; if the authentication mechanism is compromised, third parties can masquerade as the local client. For the most part, these security issues are unchanged by the introduction of an ICR, and can be dealt with using established techniques, including (but not limited to) hardware encryption, software encryption,

formal procedures for id and password management, third party security audits, tiger teams, and user education.

[0044] The introduction of an ICR adds one additional security consideration, which is the risk of having mission-critical data that exists only in a relatively transient local data store. In this regard, because the ICR hybridizes the ASP model to introduce a transient local store, it also hybridizes the security threat profile to include the local storage issues from which a pure ASP is somewhat exempt. Threats to local data security should be addressed in the manner traditional for site-based (i.e. non-ASP) applications: physical security, access controls, and usage monitoring are the key protection mechanisms available.

[0045] The logic and actions that constitute an ICR may be carried out on any number of possible computing platforms, such as a commercial or non-commercial operating system, a programmable logic unit in which all operations are embedded in "firmware", or any other engine capable of supporting the ICR logic Such an engine will require an (application-dependent) adequate amount of temporary storage area for the cached data, which may be provided via magnetic disk, non-volatile ("flash") memory, or any other rewritable storage medium.

Description of ICR Startup Process

[0046] Referring to **FIG. 1**, when an ICR is turned on or restarted, it may first seek to ascertain whether or not it is being assigned a new network identity. This may be implemented using a variety of methods, such as:

[0047] physical switches built in to an ICR implemented as a dedicated hardware system,

[0048] initialization files used by an ICR implemented on top of a standard computer operating system,

[0049] physical cues such as keyboard commands issued at power-on via hardware connected to the ICR.

[0050] If a new network identity is being assigned, the ICR ascertains and verifies its new identity before proceeding. (In an alternate embodiment, it might also be possible to change the network identity in the middle of operation.) The network identity is used by the ICR to locate the remote server whose service it is augmenting, and optionally to identify and authenticate the ICR to that remote server.

[0051] After optionally verifying a new network identity, the ICR initializes its global state variables and launches one or more processes whose collective operation implements the functionality of the ICR. These are described here as separate computing processes, or threads, for simplicity of understanding, but an alternate embodiment could combine all of these processes into a single or smaller number of threads. The ICR initializes a global variable describing the current Internet connectivity state (typically initialized to "no connection") and launches the five processes that are shown in FIGS. **2-6**.

Transaction Listener Process

[0052] The ICR Transaction Listener Process waits until the ICR receives a transaction request from one of the Thin Client terminals, much like a web server or any other network server. A transaction request may come in the form of an HTTP (web) transaction or may use any other transaction-oriented network protocol.

[0053] As shown in **FIG. 2**, when a transaction is received from the thin client, the ICR spawns a new thread, subroutine, procedure, or other process for responding to the request. It first evaluates the request to see if it is in the application subset designated as "mission-critical." If the transaction is not mission-critical, then the transaction is simply re-routed to the remote server if the Internet connection is functioning, or returns a failure code to the Thin Client terminal if the Internet is unavailable or the server is otherwise unreachable.

[0054] Several variations of this sequence are possible, including performing immediate processing of the synchronization transaction, or delaying the provision of a mission-critical result to the Thin Client until the server's answer is received in the case where the Internet is currently available.

Internet Connection Control Process

[0055] The ICR Internet Connection Control Process continuously monitors the state of the ICR's connection to the Internet, providing state information to the other processes that allows them to base their logic on the known state of the Internet without necessarily having to re-test connectivity before each operation. (In an alternate, less efficient embodiment, such testing might be done implicitly or explicitly for every network-related operation, in which case the Internet Connection Control Process would not be needed.)

[0056] As shown in **FIG. 3**, the Internet Connection Control Process wakes up periodically and checks the globally-available settings describing the current state of the Internet connection. If the Internet connection is currently believed to be functional, this process seeks to confirm that connectivity by communicating briefly with the remote server. If that communication fails, the global state is altered to indicate that no Internet connectivity is currently available.

[0057] If no connection is available, the ICR attempts to reconnect to the Internet via a plurality of configured mechanisms, which may include dedicated connections such as leased lines, DSL, cable modems, satellite connections, modems on conventional telephone lines, or wireless modems. If connectivity is restored via any of these mechanisms, the global state is updated accordingly. In any event, the Internet Connection Control Process waits for a certain amount of time and then repeats the entire process again.

Transaction Synchronization Process

[0058] The ICR Transaction Synchronization Process is responsible for making sure that all mission-critical transactions that have been processed by the ICR are synchronized, as quickly as practical, with the remote server. In normal, connected operation, this process seeks to empty its queue (and thus fully synchronize transaction state with the remote server) within a very brief interval after the transaction synchronization request is generated. However, the transaction queue will retain unprocessed synchronization requests during periods of Internet connection outages.

[0059] As shown in **FIG. 4**, the Transaction Synchronization Process wakes up periodically and checks to see if

there are any transactions are in the synchronization queue and if the Internet connection is functional. If so, it synchronizes each queued transaction by passing the original Thin Client's request to the server, comparing the server's answer with the stored answer already given by the ICR Transaction Listener Process. If no answer is received, the synchronization request is left in the queue. If the answer received differs from the stored answer differ. the ICR sends an "exception" transaction to the remote processor, informing it of the anomaly. The synchronization event is removed from the queue, and any additional queued transactions are processed.

Data Cache Control Process Logic

[0060] The ICR Data Cache Control Process is responsible for making sure that all items in the data cache—that is, all data that is necessary for mission-critical functionality—is up to date and mirrors the primary copy of such data on the remote server.

[0061] As shown in **FIG. 5**, the Data Cache Control Process simply waits for the remote server to notify it of data that needs to be updated, and then downloads that data from the remote server. Internet outages that occur during this update process simply cause the updating to be delayed until connectivity is restored.

Program Cache Control Process Logic

[0062] The ICR Program Cache Control Process is responsible for making sure that all items in the program cache—that is, all of the actual executable or interpreted programs that are necessary for mission-critical functionality—are up to date and mirror the primary copy of such programs on the remote server.

[0063] As shown in **FIG. 6**, the Program Cache Control Process periodically polls the remote server to request a list of programs that needs to be updated, and then downloads those programs from the remote server. Internet outages that occur during this update process simply cause the updating to be delayed until connectivity is restored.

Alternative Implementations and Embodiments

[0064] The logic flows just described are only one general outline of ICR logic processing. Many variations are possible. For example:

[0065] An ICR might be instantiated on a general purpose computing box, in which case various additional operating system processes might be occurring simultaneous with the logic outlined here.

[0066] Firewall and other routing functionality may be combined with the ICR logic. Thus, for example, where Transaction Listener Process step 2.a. 1 says "reroute transaction to remote server" this might actually mean redirection through the firewall component of the ICR.

[0067] The program cache and data cache control processes could be merged into a single process. They are shown here as separate processes because they have different requirements for latency and data reliability and thus might be processed on different schedules. In this example, the data cache is implemented with server notifications for changed data elements, while the program cache is implemented with periodic client-side polling. Either policy, or various other policies, might be implemented for the control of either of the two caches.

[0068] We claim:

1. In an application service provider (ASP) computing environment wherein a client interacts with a remote server over a shared network, a method of increasing transaction reliability, comprising the steps of:

maintaining a list of critical transactions;

locally caching at least certain processing capabilities associated with the application;

monitoring requests from the client to determine if a request relates to one of the critical transactions; and, if so:

processing that transaction locally and returning a response directly to the client.

2. The method of claim 1, further including the step of synchronizing the transaction with the remote server after processing the request.

3. The method of claim 2, wherein the synchronization contains both the request and the locally issued response.

4. The method of claim 1, assuming the request does not relate to a critical transaction, further including the step of transparently routing the transaction to the remote server if the network is functioning and, if not, returning a failure message to the client if the network is unavailable or if the server is otherwise inaccessible.

5. The method of claim 1, further including the step of monitoring the connectivity of the network in a background mode and, if a problem with connectivity is detected, taking one or more actions to overcome the problem.

6. The method of claim 5, wherein one of the actions used to overcome a problem associated with network connectivity includes routing traffic to an alternative network provider.

7. The method of claim 5, wherein one of the actions used to overcome a problem associated with network connectivity includes establishing communication through a backup link.

8. The method of claim 5, wherein one of the actions used to overcome a problem associated with network connectivity includes the use of an alternative communications infrastructure to notify network administrators of the problem.

9. The method of claim 1, wherein the application is associated with electronic commerce.

10. The method of claim 9, wherein the client is associated with a store having one or more point-of-sale terminals.

11. The method of claim 10, wherein sales, transactions are identified as critical, whereas functionality related to reporting, inventory data, and customer relationship or management are considered non-critical.

12. The method of claim 9, wherein the network is the internet.

13. In a network computing environment wherein a client interacts with a remote server providing access to an application, an intelligent caching router comprising:

a component containing software, hardware, or both, situated proximate to the location of the client and functioning as an interface to the network, the compo-

nent storing a list of critical transactions and at least some of the processing capabilities associated with the application,

the component being operative to perform the following functions:

a) monitor requests from the client to determine if a request relates to one of the critical transactions; and, if so:

b) process that transaction locally and returning a response directly to the client.

14. The intelligent caching router of claim 13, wherein the component is further operative to synchronize the transaction with the remote server after processing the request.

15. The intelligent caching router of claim 14, wherein the synchronization contains both the request and the locally issued response.

16. The intelligent caching router of claim 13, assuming the request does not relate to a critical transaction, the component being further operative to transparently route the transaction to the remote server if the network is; functioning and, if not, return a failure message to the client if the network is unavailable or if the server is otherwise inaccessible.

17. The intelligent caching router of claim 13, the component being further operative to monitor the connectivity of the network in a background mode and, if a problem with connectivity is detected, take one or more actions to overcome the problem.

18. The intelligent caching router of claim 17, wherein one of the actions used to overcome a problem associated with network connectivity includes routing traffic to an alternative network provider.

19. The intelligent caching router of claim 17, wherein one of the actions used to overcome a problem associated with network connectivity includes establishing communication through a backup link.

20. The intelligent caching router of claim 17, wherein one of the actions used to overcome a problem associated with network connectivity includes the use of an alternative communications infrastructure to notify network administrators of the problem.

21. The intelligent caching router of claim 17, wherein the application is associated with electronic commerce.

22. The intelligent caching router of claim 13, wherein the client is associated with a store having one or more point-of-sale terminals.

23. The intelligent caching router of claim 22, wherein sales transactions are identified as critical, whereas functionality related to reporting, inventory data, and customer relationship or management are considered non-critical.

24. The intelligent caching router of claim 13, wherein the network is the Internet.

25. The intelligent caching router of claim 13, further including routing or firewall functionality associated with the application.

26. The intelligent caching router of claim 13, wherein the component is further operative to perform DNS (domain name system) lookup, DHCP (dynamic host configuration protocol) service, and any other network services that are essential for the functioning of the local client.

27. In an application service provider (ASP) computing environment wherein a client interacts with a remote server over a shared network, the improvement comprising:

an intelligent caching router (ICR) inserted functionally between the client and the network, such that when conventional backup routing, fails, the ICR begins to act as a surrogate for the unreachable remote server on which the application service depends.

* * * * *

US 20020128916A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2002/0128916 A1

Beinecke, III (43) Pub. Date: Sep. 12, 2002

(54) METHODS, APPARATUS AND ARTICLES-OF-MANUFACTURE FOR DISTRIBUTING/REDEEMING A UNIVERSAL INCENTIVE CURRENCY

(76) Inventor: Walter Beinecke III, Manchester, MA (US)

Correspondence Address:
David Garrod, Ph.D., Esq.
Hopgood, Calimafde, Judlowe & Mondolino, LLP
60 East 42nd Street
New York, NY 10165 (US)

(21) Appl. No.: 09/795,893

(22) Filed: Feb. 28, 2001

Related U.S. Application Data

(60) Provisional application No. 60/185,303, filed on Feb. 28, 2000.

Publication Classification

(51) Int. Cl.[7] G06F 17/60

(52) U.S. Cl. 705/26

(57) ABSTRACT

The invention relates to an Internet-accessible incentive marketing program, which provides a plurality of html pages viewable by Internet-connected users using browser software. This program provides a plurality of services that allow Internet-connected users to accrue and utilize incentive points. Generally, a homepage is provided, including various links, such as (i) an enrollment link, for initiating an on-line enrollment process, (ii) an informational link, for providing information concerning the incentive marketing program, (iii) a login link, for accessing user account information, (iv) a time-limited incentive link, for accessing a time-limited incentive offer, (v) an incentive-referral enrollment link, for directing the user to enroll in a third-party program from which the user can accrue incentive points, (vi) a direct-access shopping link, for permitting the user to transact on-line business with an associated merchant or service provider, and (vii) a direct-access redemption link, including an associated depiction of an award and an incentive point total needed to redeem the award, for permitting the user to redeem the depicted award. The third-party program associated with an incentive-referral enrollment link may, for example, relate to Internet access services, long-distance telephone services, on-line auction services, or other services.

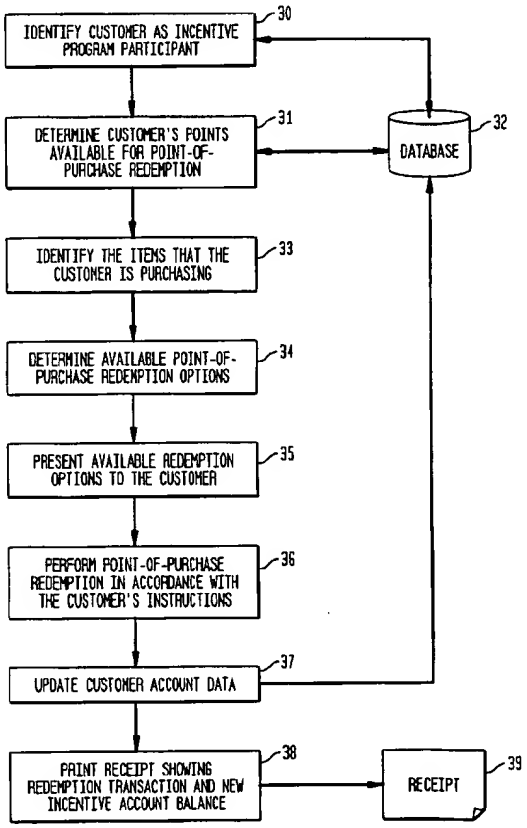


FIG. 1

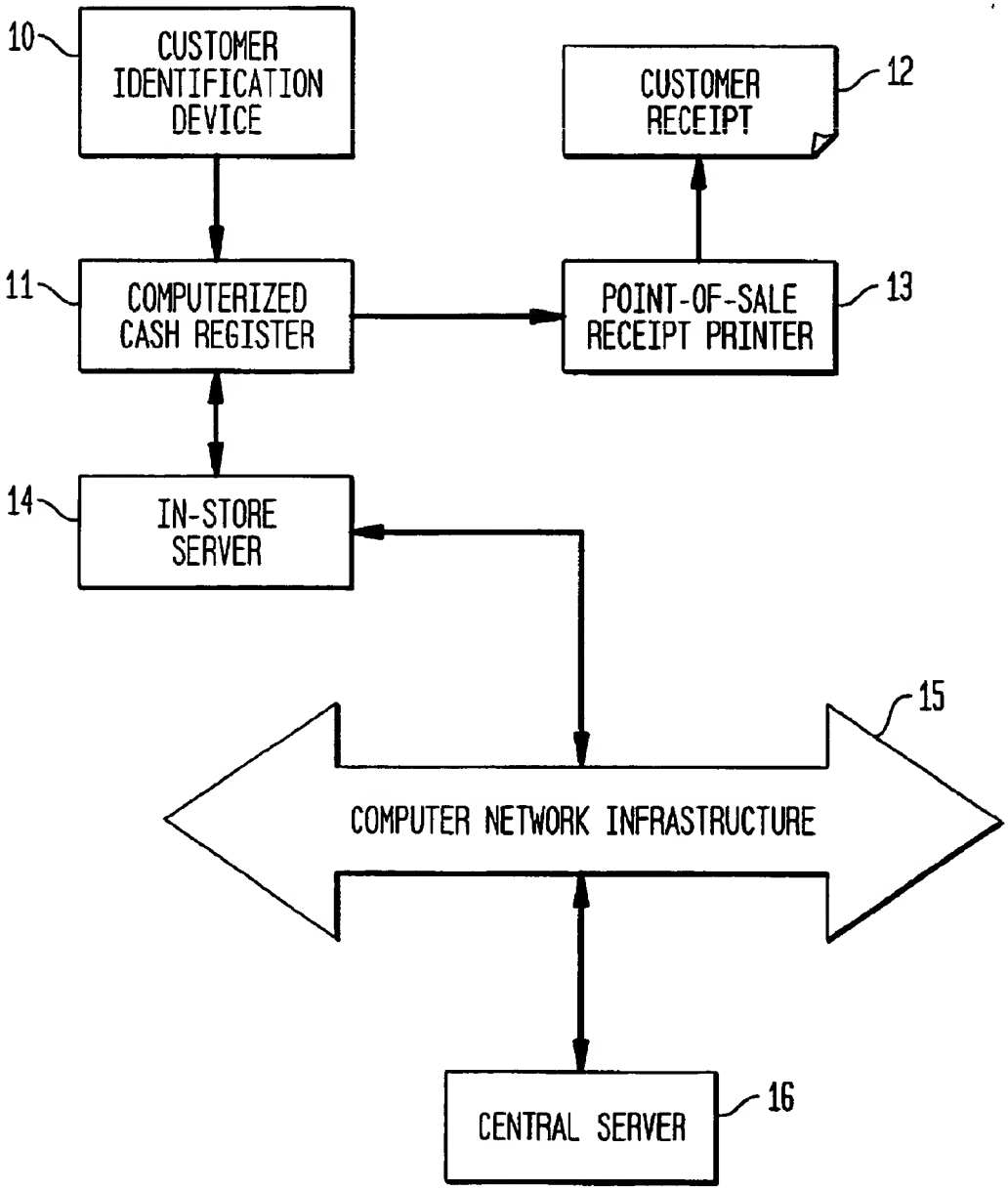

10
CUSTOMER IDENTIFICATION DEVICE
12
CUSTOMER RECEIPT
11
COMPUTERIZED CASH REGISTER
13
POINT-OF-SALE RECEIPT PRINTER
14
IN-STORE SERVER
15
COMPUTER NETWORK INFRASTRUCTURE
16
CENTRAL SERVER

FIG. 2

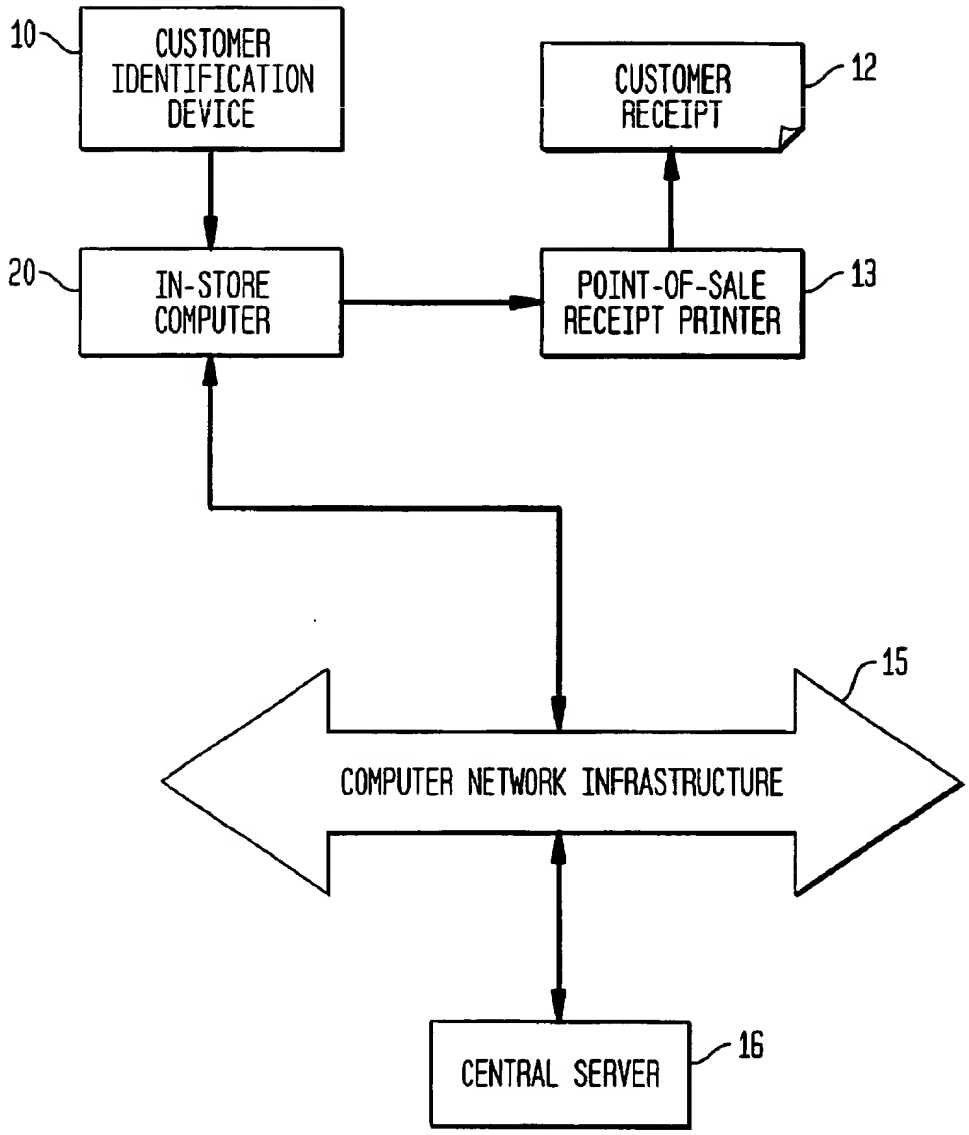

10
CUSTOMER IDENTIFICATION DEVICE
12
CUSTOMER RECEIPT
20
IN-STORE COMPUTER
13
POINT-OF-SALE RECEIPT PRINTER
15
COMPUTER NETWORK INFRASTRUCTURE
16
CENTRAL SERVER

FIG. 3A

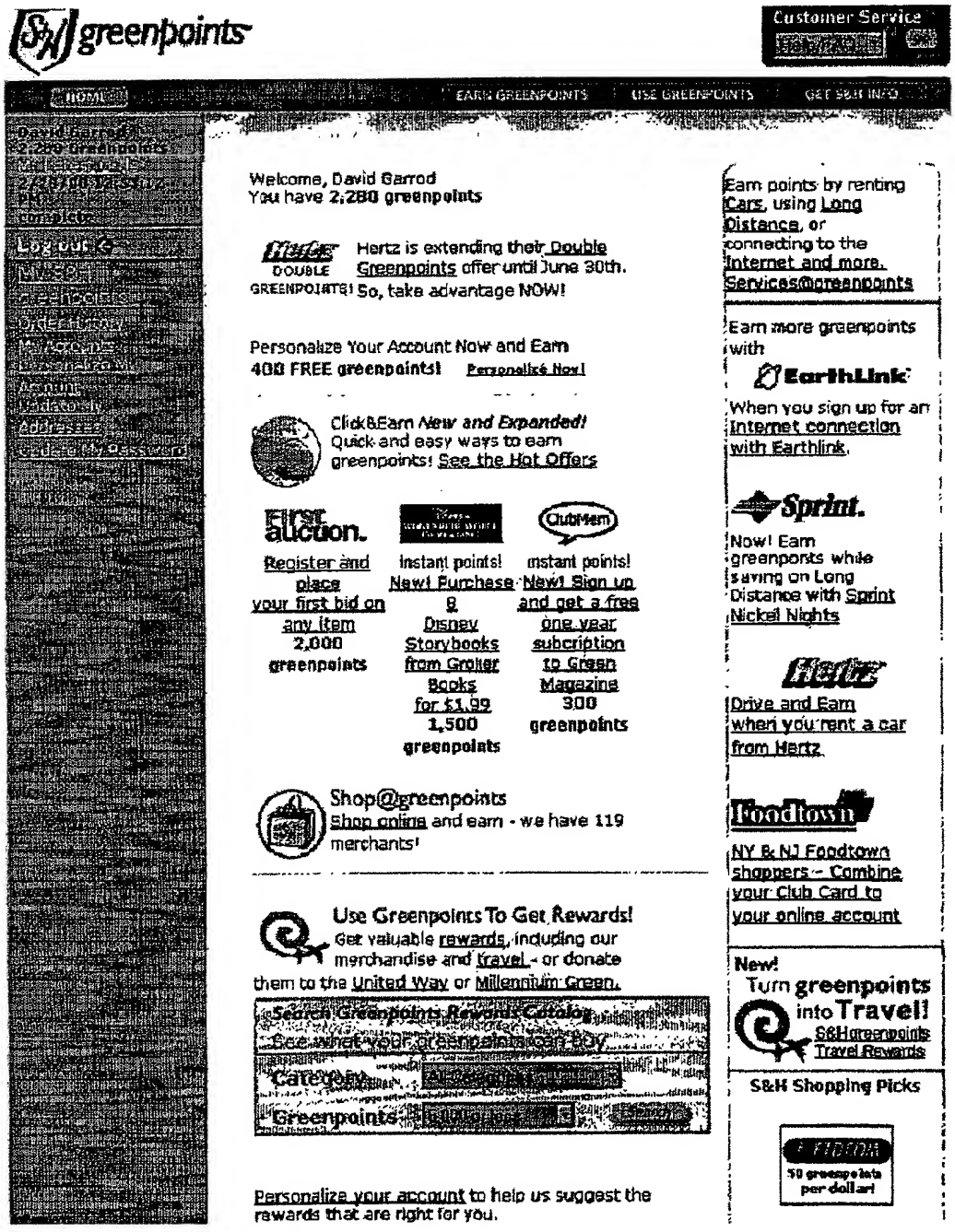

greenpoints
Customer Service
HOME
EARN GREENPOINTS
USE GREENPOINTS
GET S&H INFO
Log out
Welcome, David Garrod
You have 2,280 greenpoints
Hertz is extending their Double Greenpoints offer until June 30th. So, take advantage NOW!
Personalize Your Account Now and Earn
400 FREE greenpoints!
Personalize Now!
Click&Earn New and Expanded!
Quick and easy ways to earn greenpoints! See the Hot Offers
First auction.
Register and place your first bid on any item
2,000 greenpoints
Instant points! New! Purchase 8 Disney Storybooks from Grolier Books for $1.99
1,500 greenpoints
ClubMom
Instant points! New! Sign up and get a free one year subcription to Green Magazine
300 greenpoints
Shop@greenpoints
Shop online and earn - we have 119 merchants!
Use Greenpoints To Get Rewards!
Get valuable rewards, including our merchandise and travel - or donate them to the United Way or Millennium Green.
Search Greenpoints Rewards Catalog
See what your greenpoints can buy
Category
Greenpoints
Personalize your account to help us suggest the rewards that are right for you.
Earn points by renting Cars, using Long Distance, or connecting to the Internet and more. Services@greenpoints
Earn more greenpoints with
EarthLink
When you sign up for an Internet connection with Earthlink.
Sprint.
Now! Earn greenpoints while saving on Long Distance with Sprint Nickel Nights
Hertz
Drive and Earn when you rent a car from Hertz.
Foodtown
NY & NJ Foodtown shoppers - Combine your Club Card to your online account
New!
Turn greenpoints into Travel!
S&Hgreenpoints Travel Rewards
S&H Shopping Picks
50 greenpoints per dollar!

FIG. 3B

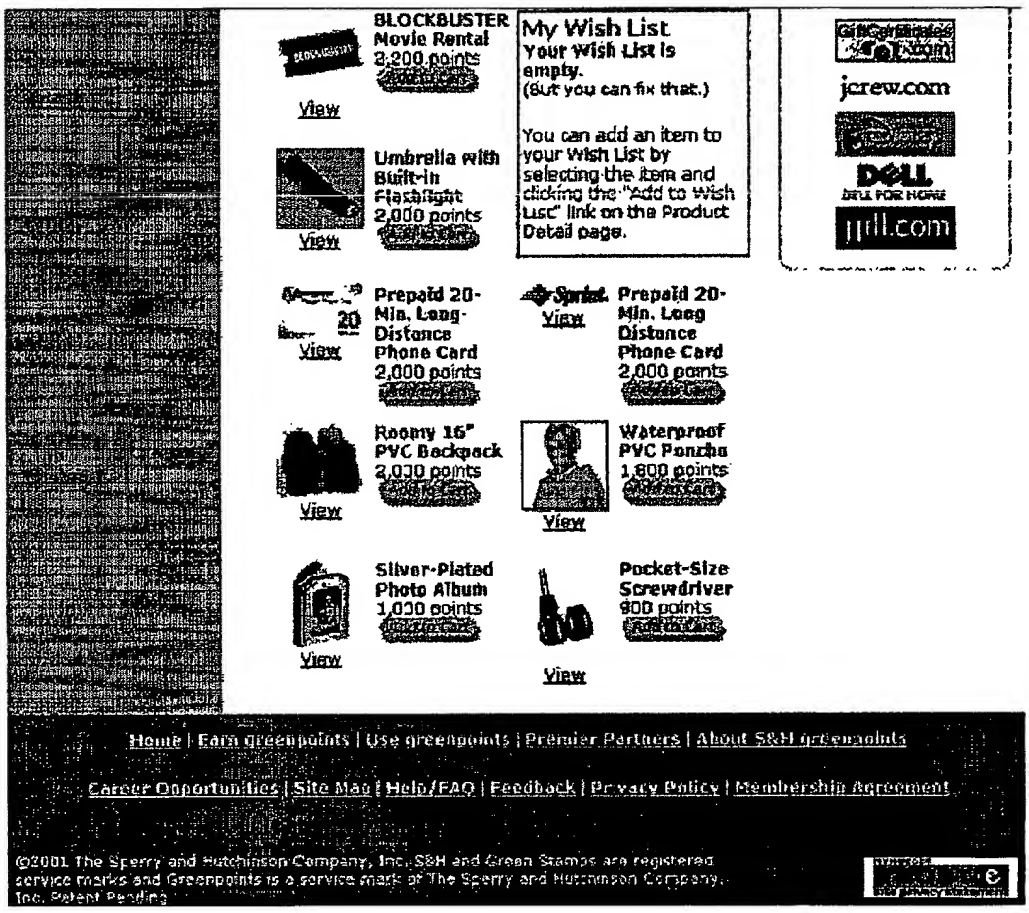

BLOCKBUSTER
Movie Rental
3,200 points
View
My Wish List
Your Wish List is empty.
(But you can fix that.)
You can add an item to your Wish List by selecting the item and clicking the "Add to Wish List" link on the Product Detail page.
jcrew.com
DELL FOR HOME
jrll.com
Umbrella with Built-in Flashlight
2,000 points
View
Prepaid 20-Min. Long-Distance Phone Card
2,000 points
View
Prepaid 20-Min. Long Distance Phone Card
2,000 points
View
Roomy 16" PVC Backpack
2,000 points
View
Waterproof PVC Poncho
1,800 points
View
Silver-Plated Photo Album
1,000 points
View
Pocket-Size Screwdriver
900 points
View
Home | Earn greenpoints | Use greenpoints | Premier Partners | About S&H greenpoints
Career Opportunities | Site Map | Help/FAQ | Feedback | Privacy Policy | Membership Agreement
©2001 The Sperry and Hutchinson Company, Inc. S&H and Green Stamps are registered service marks and Greenpoints is a service mark of The Sperry and Hutchinson Company, Inc. Patent Pending

FIG. 4

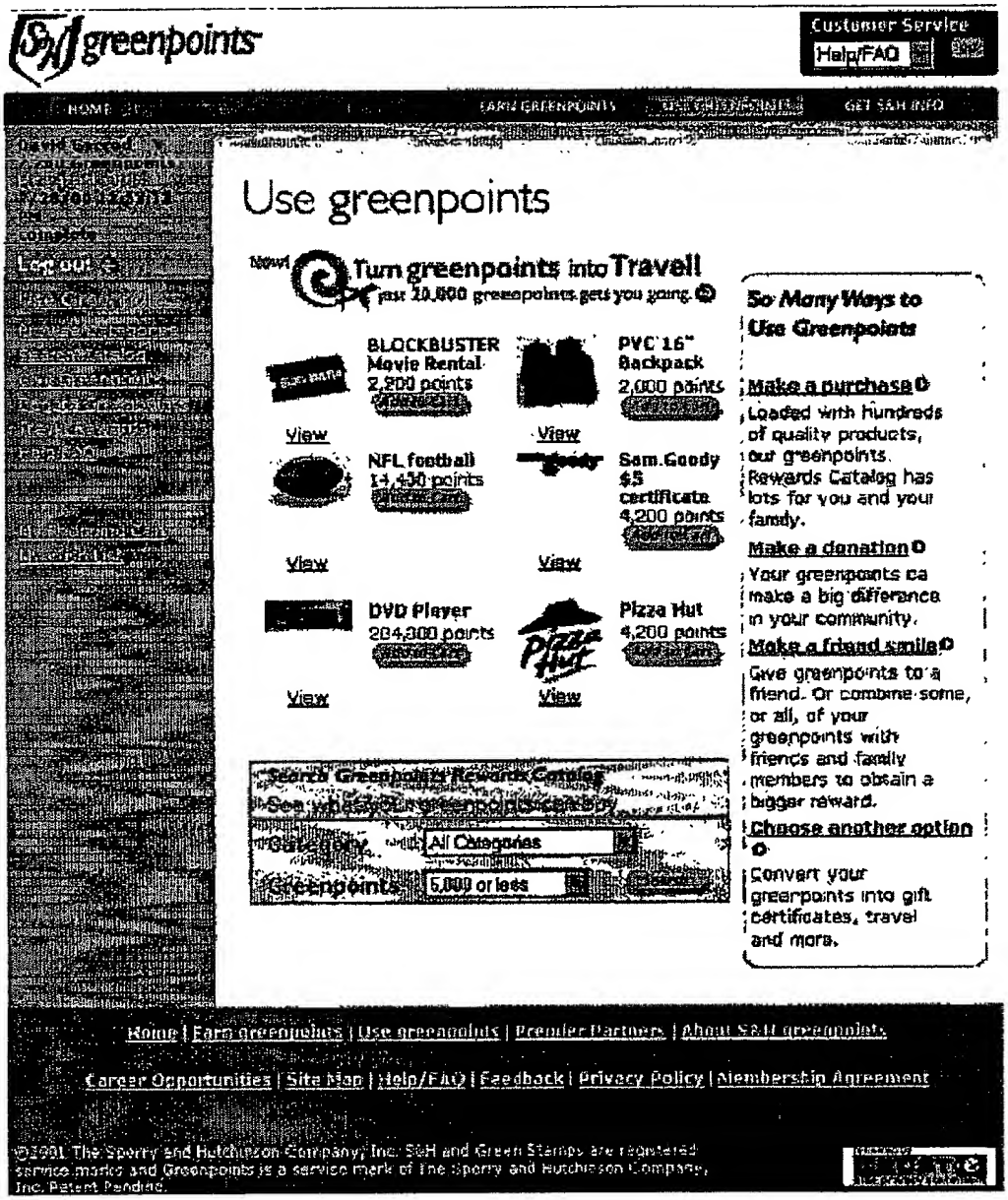
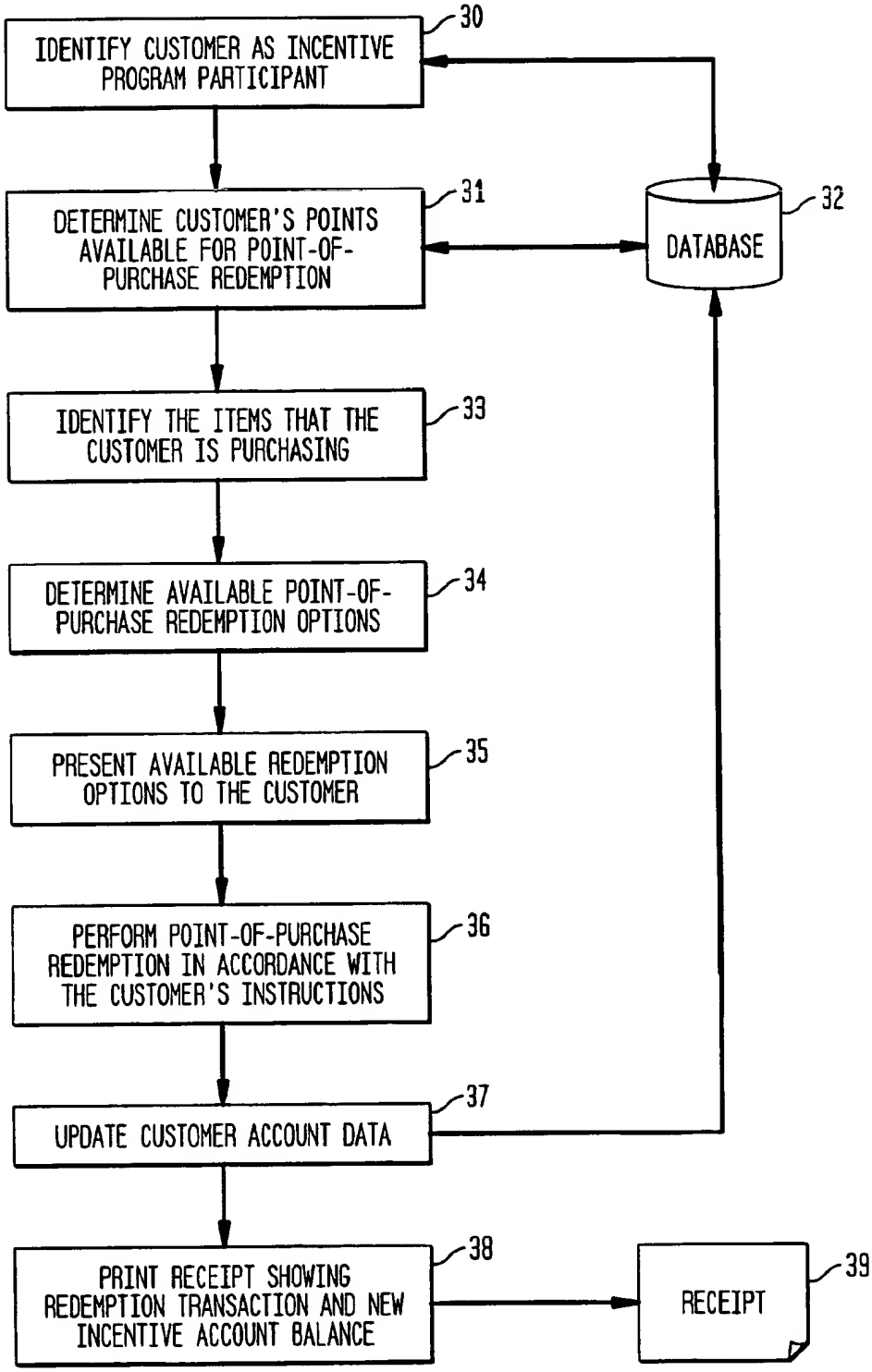

greenpoints
Customer Service
Help/FAQ
Use greenpoints
Turn greenpoints into Travel!
BLOCKBUSTER Movie Rental
2,200 points
PVC 16" Backpack
2,000 points
View
NFL football
14,450 points
Sam Goody $5 certificate
4,200 points
DVD Player
284,300 points
Pizza Hut
4,200 points
All Categories
5,000 or less
So Many Ways to Use Greenpoints
Make a purchase
Loaded with hundreds of quality products, our greenpoints Rewards Catalog has lots for you and your family.
Make a donation
Your greenpoints ca make a big difference in your community.
Make a friend smile
Give greenpoints to a friend. Or combine some, or all, of your greenpoints with friends and family members to obtain a bigger reward.
Choose another option
Convert your greenpoints into gift certificates, travel and more.
Home | Earn greenpoints | Use greenpoints | Premier Partners | About S&H greenpoints
Career Opportunities | Site Map | Help/FAQ | Feedback | Privacy Policy | Membership Agreement

FIG. 5


IDENTIFY CUSTOMER AS INCENTIVE PROGRAM PARTICIPANT
30
DETERMINE CUSTOMER'S POINTS AVAILABLE FOR POINT-OF-PURCHASE REDEMPTION
31
DATABASE
32
IDENTIFY THE ITEMS THAT THE CUSTOMER IS PURCHASING
33
DETERMINE AVAILABLE POINT-OF-PURCHASE REDEMPTION OPTIONS
34
PRESENT AVAILABLE REDEMPTION OPTIONS TO THE CUSTOMER
35
PERFORM POINT-OF-PURCHASE REDEMPTION IN ACCORDANCE WITH THE CUSTOMER'S INSTRUCTIONS
36
UPDATE CUSTOMER ACCOUNT DATA
37
PRINT RECEIPT SHOWING REDEMPTION TRANSACTION AND NEW INCENTIVE ACCOUNT BALANCE
38
RECEIPT
39

METHODS, APPARATUS AND ARTICLES-OF-MANUFACTURE FOR DISTRIBUTING/REDEEMING A UNIVERSAL INCENTIVE CURRENCY

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application derives from U.S. Provisional Patent Application S/N 60/185,303, filed Feb. 28, 2000, which prior provisional application is hereby incorporated by reference.

FIELD OF THE INVENTION

[0002] The present invention relates generally to the field of computer-assisted business methods, and to systems and articles-of-manufacture for implementing such methods. More particularly, the invention relates to computer-based methods, apparatus and articles-of-manufacture for implementing and/or operating loyalty rewards programs, and the like.

BACKGROUND OF THE INVENTION

[0003] Incentive marking and loyalty rewards programs have existed for many years. An example is the well-known S&H Green Stamps program, wherein participants are awarded "Green Stamps" by participating merchants, in exchange for purchases or other qualifying activities. Once a participant collects a sufficient number of Green Stamps, he/she may redeem the Stamps for rewards, such as free merchandise, or other valuable consideration. Another familiar example of a loyalty rewards program is the "frequent flyer" program, wherein an airline awards points for flights on the particular airline, which points may eventually be redeemed for free travel on the particular airline.

[0004] With the widespread promulgation of computer technology in the business world, a variety of computer-implemented loyalty rewards programs have been reported, in recent years. For example, U.S. Pat. No. 5,483,444, SYSTEM FOR AWARDING CREDITS TO PERSONS WHO BOOK TRAVEL-RELATED RESERVATIONS, describes a system for rewarding agents and other persons, based on their on-line travel and/or hotel bookings. U.S. Pat. No. 5,025,372, SYSTEM AND METHOD FOR ADMINISTRATION OF INCENTIVE AWARD PROGRAM THROUGH USE OF CREDIT, discloses a system for rewarding credit card users by issuing "kickbacks" to the card-holder's account, with the size of the kickbacks being dependent on the amount of credit card use or interest paid by the customer. U.S. Pat. No. 5,710,887, COMPUTER SYSTEM AND METHOD FOR ELECTRONIC COMMERCE, describes a system that awards "frequent buyer points" to customers of an on-line shopping mall. As a final example, U.S. Pat. No. 5,056,019, AUTOMATED PURCHASE REWARD ACCOUNTING SYSTEM AND METHOD, discloses a system in which participants are credited with incentive reward points, for in-store purchases, through use of a bar-coded membership card which identifies the participant.

[0005] The above-mentioned '444, '372, '887 and '019 patents, each of which is incorporated herein by reference, evidence a current state-of-the-art in which incentive credits can be awarded, maintained and redeemed by computer, and in which participants in the incentive program can communicate with the computer (via, for example, the Internet) to check incentive point balances and redeem points for selected awards.

[0006] Despite the wide promulgation of computer-based loyalty-rewards systems, present-day systems suffer from a number of deficiencies that diminish their market acceptance and reduce their loyalty-influencing power. For example, present-day loyalty-rewards systems typically award incentive credits for a specific type of activity, such as on-line purchases at a particular web site, use of a specific credit or debit card, in-store purchases at particular stores, travel on particular carriers, etc. As a result, a consumer interested in taking maximum advantage of available incentives is forced to enroll in numerous programs, remember which program covers which activity (e.g., the "X" program covers in-store purchases, the "Y" program covers on-line purchases, etc.), and remember how and when to redeem the points earned in each of the various programs. This presents a frustrating situation to the average consumer, and is believed (by the inventor herein) to be a major impediment to the widespread use of loyalty rewards programs. Thus, there remains an unfulfilled need for a computer-based incentive program that better meets the requirements of consumers, yet provides all of the conveniences of today's fully-automated, on-line systems.

OBJECTS AND DESCRIPTION OF THE INVENTION

[0007] In light of the above, one general object of the present invention is a computer-based method, apparatus and article-of-manufacture for implementing and/or operating a unified incentive/loyalty rewards system.

[0008] Another general object of the invention is a computer-based method, apparatus and article-of-manufacture for implementing and/or operating an incentive/loyalty rewards system in which participants can earn points from a wide variety of qualifying activities.

[0009] Yet another general object of the invention is a computer-based method, apparatus and article-of-manufacture for implementing and/or operating an incentive/loyalty rewards system in which the qualifying activities, for which participants are awarded points, include both off-line (e.g., in the store) purchases and on-line (e.g., through the Internet) purchases.

[0010] A further general object of the invention is a computer-based method, apparatus and article-of-manufacture for implementing and/or operating an incentive/loyalty rewards system which permits users to aggregate and assign their points to each other (e.g., a user may assign some or all of his/her points to another user, or to a group) and/or to charitable organizations (e.g., a school or national charity).

[0011] A still further general object of the invention is a computer-based method, apparatus and article-of-manufacture for implementing and/or operating an incentive/loyalty rewards system which permits users to redeem their points either on-line or off-line.

[0012] Still further objects of the invention relate to methods, apparatus and/or articles-of-manufacture to facilitate collection of customer behavior information when custom-

ers engage in incentive-accruing transactions or activities, and to utilize such information to target promotional offers.

[0013] These, and other objects/advantages, are realized by the present invention, which preferably comprises at least one server, connected to a computer network. The invention may also include a plurality of additional servers, to service such functions as offline redemption and/or regional points collection and accounting functions. Points may be earned for a wide variety of qualifying activities, including, but not limited to, on-line purchases, offline purchases, flights, hotel stays, credit/debit card use, participation in surveys, telephone use, banking activity, gasoline purchases, internet site visits, and on-line contests or lotteries. For purposes of redemption and/or qualification, users may be identified by a variety of means, including, but not limited to, on-line signatures (e.g., user id/password or cryptographic id), bar-coded identification cards, magnetic cards, smart cards, or biometric identifiers (e.g., fingerprint or iris pattern).

[0014] Accordingly, generally speaking, and without intending to be limiting, one aspect of the invention relates to a method for operating an Internet-accessible incentive marketing program, comprising the following: providing, via an Internet-accessible server, a plurality of html pages, including a homepage, the html pages viewable by Internet-connected users using browser software; providing, through the Internet-accessible server, a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; providing, on the homepage, at least the following: (i) an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process; (ii) an informational link, selectable by an Internet-connected user to provide information concerning the incentive marketing program; (iii) a login link, selectable by an Internet-connected user to access user account information; and (iv) at least two of the following: (a) at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer; (b) at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct the user to enroll in a third-party program from which the user can accrue incentive points; (c) a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and (d) a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award. The third-party program associated with the at least one incentive-referral enrollment link may relate to Internet access services, long-distance telephone services, on-line auction services, or other services.

[0015] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a method for operating an Internet-accessible incentive marketing program, comprising: providing, via an Internet-accessible server, a plurality of html pages, including a homepage, the html pages viewable by Internet-connected users using browser software; providing, through the Internet-accessible server, a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; and providing, on the homepage, a brick-and-mortar customer referral link, selectable by an Internet-connected user to permit the user to utilize incentive points earned in connection with brick-and-mortar purchases. The method may further involve providing, on the homepage, at least one, two, three, or all of the following: (i) at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer; (ii) at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct the user to enroll in a third-party program from which the user can accrue incentive points; (iii) a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and/or (iv) a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0016] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a method for operating an Internet-accessible incentive point redemption system, comprising: maintaining, on an Internet-accessible server, data corresponding to a multiplicity of customer accounts; providing an Internet-accessible homepage through which an Internet-connected user can access the server to interrogate or utilize his/her account; providing, on the Internet-accessible homepage, a primary redemption link, selectable by an Internet-connected user to navigate the user to a redemption homepage; providing, on the redemption homepage, at least one, two, three or more of the following: (i) a charitable donation link, selectable by an Internet-connected user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations; (ii) an assignment link, selectable by an Internet-connected user to initiate a process whereby the user can assign incentive points from his/her account to another user's account; (iii) a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services; (iv) an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (a) points required for redemption, (b) category of reward, or both (a) and (b); and (v) a plurality of direct-access redemption links, each including an associated depiction of an award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0017] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a method for operating an Internet-accessible incentive marketing program, comprising: providing, via an Internet-accessible server, a plurality of Web pages viewable by Internet-connected users using browser software; providing, through the Internet-accessible server, a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; providing a Web homepage, having at least the following: (i) an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process; (ii) an informational link, selectable by an Internet-connected user to provide information concerning the incentive marketing program; (iii) a login link, selectable by an Internet-connected user to access

user account information; (iv) a redemption homepage link, selectable by an Internet-connected user to access a redemption homepage; and (v) at least one, two, three or more of the following: (a) at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer; (b) at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable the user to enroll in a third-party program from which the user can accrue incentive points; (c) at least one direct-access shopping link, selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and (d) at least one direct-access redemption link, the direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit the user to redeem the depicted award; and providing, on the redemption homepage, at least one, two, three or more of the following: (a) a charitable donation link, selectable by an Internet-connected user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations; (b) an assignment link, selectable by an Internet-connected user to initiate a process whereby the user can assign incentive points from his/her account to another user's account; (c) a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services; (d) an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and (e) a plurality of direct-access redemption links, each including an associated depiction of an award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0018] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a method for operating an incentive marketing program, comprising: maintaining, in a database, a list of enrolled members and account information for each member, the account information including a current incentive point balance; identifying a customer as an enrolled member; using the database to determine, in real time, the identified customer's incentive point balance; using the identified customer's incentive point balance to determine whether the customer is eligible for a discount on goods being purchased by the customer and, if so, offering a discount to the customer in exchange for redemption of all or part of the customer's incentive point balance; and, in response to the customer's directive, redeeming all or part of the customer's incentive point balance to provide a purchase discount to the customer, updating the database to reflect the redemption, and providing a receipt to the customer that details the redemption and the purchase discount. Identifying a customer as an enrolled member may involve comparing data received from a magnetic stripe reader at the point-of-purchase to data stored in the database, comparing data received from an optical bar code reader at the point-of-purchase to data stored in the database, comparing data received from a smart card reader at the point-of-purchase to data stored in the database, or comparing data entered into a data entry device at the point-of-purchase to data stored in the database.

[0019] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a method for operating an incentive marketing program, comprising: maintaining, in a database, a list of enrolled members and account information for each member, the account information including a current incentive point balance; identifying a customer, at a point-of-purchase location, as an enrolled member; using the database to determine, in real time, the identified customer's incentive point balance; determining whether any items that the identified customer has presented for purchase can be acquired or discounted through redemption of incentive point's in the customer's account and, if so, offering to redeem incentive points in exchange for discount(s) on, and/or acquisition(s)of, one or more items that the customer has presented for purchase.

[0020] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a method of operating a universal incentive marketing program that permits registered users to accrue incentive points for both on-line and in-store transactions, comprising: maintaining a network-accessible central server that includes a database of account information for registered users; providing a customer identification device at a retail point-of-purchase location that permits identification of a customer as a registered user; adding incentive points to the accounts of customers identified as registered users for in-store purchases made by the users; providing an Internet-accessible homepage linked to the central server to permit identification of an Internet-connected user as a registered user, the homepage further including at least one, two or more of the following: (a) at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer; (b) at least one incentive-referral enrollment link, selectable by the user to direct the user to enroll in a third-party program from which the user can accrue incentive points; (c) a plurality of direct-access shopping links, each selectable by the user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and (d) a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by the user to permit the user to redeem the depicted award. The homepage may also include a link to a redemption homepage, and the redemption homepage includes at least one, two or more of the following: (a) a charitable donation link, selectable by the user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations; (b) an assignment link, selectable by the user to initiate a process whereby the user can assign incentive points from his/her account to another user's account; (c) a travel redemption link, selectable by the user to initiate a process whereby incentive points can be redeemed for travel services; (d) an awards catalog search tool, engageable by the user to search a catalog of awards that can be obtained using incentive points; and (e) a plurality of direct-access redemption links, each including an associated depiction of an award, the direct-access redemption links each selectable by the user to permit the user to redeem the depicted award.

[0021] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a methods, apparatus, or articles-of-manufacture for operating an incentive marketing program, and using collected customer data to target customer promotions by, for example, the following: maintaining, in a database, a list of enrolled

members of said incentive marketing program; identifying, in real time, at a point-of-purchase location, a customer as an enrolled member; and determining, in real time, whether said customer is eligible to participate in a promotion and, if so, presenting the promotion to the customer. The promotion may involve one or more of: a discount on product(s) that the customer is purchasing; a discount on product(s) that the customer is purchasing, contingent upon the customer purchasing selected additional product(s); a pre-approved credit card offer; a discount travel offer; an offer to earn incentive points at an accelerated rate; and/or an offer to earn free merchandise if the customer makes additional purchases, satisfies certain conditions, or performs certain actions.

[0022] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to an Internet-accessible incentive marketing system, comprising: an Internet-accessible server, configured to serve a plurality of html pages, including a homepage, viewable by Internet-connected users using browser software; the server further configured to provide a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; the homepage including a brick-and-mortar customer referral link, selectable by an Internet-connected user to permit the user to utilize incentive points earned in connection with brick-and-mortar purchases. The homepage may further include at least one, two or more of the following: (a) at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer; (b) at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct the user to enroll in a third-party program from which the user can accrue incentive points; (c) a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and (d) a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0023] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a universal incentive marketing system that permits registered users to accrue incentive points for both on-line and in-store transactions, comprising: a network-accessible central server that includes a database of account information for registered users; a customer identification device, comprising one of a bar code reader, a magnetic stripe reader, a smart card reader, or a manual data entry device, the customer identification device permitting identification of a customer as a registered user; an incentive point accounting module, configured to add incentive points to the accounts of customers identified as registered users for in-store purchases made by the users; and an Internet-accessible homepage linked to the central server to permit identification of an Internet-connected user as a registered user, the homepage further including at least one, two or more of the following: (a) at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer; (b) at least one incentive-referral enrollment link, selectable by the user to direct the user to enroll in a third-party program from which the user can accrue incentive points; (c) a plurality of direct-access shopping links, each selectable by the user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and (d) a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by the user to permit the user to redeem the depicted award. The homepage may also include a link to a redemption homepage, and the redemption homepage may include at least one, two or more of the following: (a) a charitable donation link, selectable by the user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations; (b) an assignment link, selectable by the user to initiate a process whereby the user can assign incentive points from his/her account to another user's account; (c) a travel redemption link, selectable by the user to initiate a process whereby incentive points can be redeemed for travel services; (d) an awards catalog search tool, engageable by the user to search a catalog of awards that can be obtained using incentive points; and (e) a plurality of direct-access redemption links, each including an associated depiction of an award, the direct-access redemption links each selectable by the user to permit the user to redeem the depicted award.

[0024] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to an Internet-accessible incentive marketing system, comprising: an Internet-accessible server, configured to serve up a plurality of html pages, including a homepage, the html pages viewable by Internet-connected users using browser software; the Internet-accessible server further configured to serve up a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; the homepage including at least the following: (a) an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process; (b) an informational link, selectable by an Internet-connected user to provide information concerning the incentive marketing program; (c) a login link, selectable by an Internet-connected user to access user account information; and (d) at least one, two or more of the following: (e) at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer; (f) at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct the user to enroll in a third-party program from which the user can accrue incentive points; (g) a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and (h) a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award. The at least one incentive-referral enrollment link may relate to Internet access services, long-distance telephone services, on-line auction services, or other services.

[0025] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to a method for operating an incentive marketing system, comprising: a database containing a list of enrolled members and account information for each member, the account informa-

tion including a current incentive point balance; a customer identification device, located at a point-of-purchase, and configured to identify a customer as an enrolled member; a customer information query module, configured to query the database to determine, in real time, the identified customer's incentive point balance; and a point-of-purchase redemption processing module, configured to determine, in real time, whether any items that the identified customer has presented for purchase can be acquired or discounted through redemption of incentive point's in the customer's account and, if so, to present an offer to redeem incentive points in exchange for discount(s) on, and/or acquisition(s)of, one or more items that the customer has presented for purchase.

[0026] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to an Internet-accessible incentive point redemption system, comprising: an Internet-accessible server that stores data corresponding to a multiplicity of customer accounts; an Internet-accessible homepage through which an Internet-connected user can access the server to interrogate or utilize his/her account; the Internet-accessible homepage including a primary redemption link, selectable by an Internet-connected user to navigate the user to a redemption homepage; the redemption homepage including at least one, two or more of the following: (a) a charitable donation link, selectable by an Internet-connected user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations; (b) an assignment link, selectable by an Internet-connected user to initiate a process whereby the user can assign incentive points from his/her account to another user's account; (c) a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services; (d) an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii); and (e) a plurality of direct-access redemption links, each including an associated depiction of an award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0027] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to an incentive marketing system, comprising: a database that includes a list of enrolled members and account information for each member, the account information including a current incentive point balance; means for identifying a customer as an enrolled member; means for using the database to determine, in real time, the identified customer's incentive point balance; means for using the identified customer's incentive point balance to determine whether the customer is eligible for a discount on goods being purchased by the customer and, if so, offering a discount to the customer in exchange for redemption of all or part of the customer's incentive point balance; and means, responsive to the customer's directive, for redeeming all or part of the customer's incentive point balance to provide a purchase discount to the customer, updating the database to reflect the redemption, and providing a receipt to the customer that details the redemption and the purchase discount. The means for identifying a customer as an enrolled member may comprise means for comparing data received from a magnetic stripe reader at the point-of-purchase to data stored in the database, means for comparing data received from an optical bar code reader at the point-of-purchase to data stored in the database, means for comparing data received from a smart card reader at the point-of-purchase to data stored in the database, or means for comparing data entered into a data entry device at the point-of-purchase to data stored in the database.

[0028] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to an Internet-accessible incentive marketing system, comprising: an Internet-accessible server, configured to serve up a plurality of Web pages viewable by Internet-connected users using browser software; the Internet-accessible server further configured to serve up a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; a Web homepage, having at least some of the following: (a) an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process; (b) an informational link, selectable by an Internet-connected user to provide information concerning the incentive marketing program; (c) a login link, selectable by an Internet-connected user to access user account information; (d) a redemption homepage link, selectable by an Internet-connected user to access a redemption homepage; and (e) at least one, two or more of the following: (f) at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer; (g) at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable the user to enroll in a third-party program from which the user can accrue incentive points; (h) at least one direct-access shopping link, selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and (i) at least one direct-access redemption link, the direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit the user to redeem the depicted award; and a redemption homepage that includes at least one, two or more of the following: (a) a charitable donation link, selectable by an Internet-connected user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations; (b) an assignment link, selectable by an Internet-connected user to initiate a process whereby the user can assign incentive points from his/her account to another user's account; (c) a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services; (d) an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and (e) a plurality of direct-access redemption links, each including an associated depiction of an award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0029] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to article(s)-of-manufacture for operating an incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to: maintain, in a database, a list of enrolled members and account information for each member, the account information including a current incentive point balance; identify a customer, at a point-of-purchase location, as an enrolled member; use the database to

determine, in real time, the identified customer's incentive point balance; determine, in real time, whether any items that the identified customer has presented for purchase can be acquired or discounted through redemption of incentive point's in the customer's account and, if so, offer to redeem incentive points in exchange for discount(s) on, and/or acquisition(s)of, one or more items that the customer has presented for purchase.

[0030] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to article(s)-of-manufacture for operating an Internet-accessible incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to: provide, via an Internet-accessible server, a plurality of html pages, including a homepage, the html pages viewable by Internet-connected users using browser software; provide, through the Internet-accessible server, a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; provide, on the homepage, at least the following: (a) an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process; (b) an informational link, selectable by an Internet-connected user to provide information concerning the incentive marketing program; (c) a login link, selectable by an Internet-connected user to access user account information; and (d) at least two, three or more of the following: (e) at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer; (f) at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct the user to enroll in a third-party program from which the user can accrue incentive points; (g) a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and (h) a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0031] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to article(s)-of-manufacture for operating a universal incentive marketing program that permits registered users to accrue incentive points for both on-line and in-store transactions, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to: maintain a network-accessible central server that includes a database of account information for registered users; receive data from a customer identification device at a retail point-of-purchase location, so as to permit identification of a customer as a registered user; add incentive points to the accounts of customers identified as registered users for in-store purchases made by the users; and provide an Internet-accessible homepage linked to the central server to permit identification of an Internet-connected user as a registered user, the homepage further including at least two of the following: (a) at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer; (b) at least one incentive-referral enrollment link, selectable by the user to direct the user to enroll in a third-party program from which the user can accrue incentive points; (c) a plurality of direct-access shopping links, each selectable by the user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and (d) a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by the user to permit the user to redeem the depicted award.

[0032] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to article(s)-of-manufacture for operating an Internet-accessible incentive point redemption system, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to: maintain, on an Internet-accessible server, data corresponding to a multiplicity of customer accounts; provide an Internet-accessible homepage through which an Internet-connected user can access the server to interrogate or utilize his/her account; provide, on the Internet-accessible homepage, a primary redemption link, selectable by an Internet-connected user to navigate the user to a redemption homepage; provide, on the redemption homepage, at least one, two or more of the following: (a) a charitable donation link, selectable by an Internet-connected user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations; (b) an assignment link, selectable by an Internet-connected user to initiate a process whereby the user can assign incentive points from his/her account to another user's account; (c) a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services; (d) an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii); and (e) a plurality of direct-access redemption links, each including an associated depiction of an award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0033] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to article(s)-of-manufacture for operating an Internet-accessible incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to: provide, via an Internet-accessible server, a plurality of html pages, including a homepage, the html pages viewable by Internet-connected users using browser software; provide, through the Internet-accessible server, a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; and provide, on the homepage, a brick-and-mortar customer referral link, selectable by an Internet-connected user to permit the user to utilize incentive points earned in connection with brick-and-mortar purchases.

[0034] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to article(s)-of-manufacture for operating an incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to: maintain, in a database, a list of enrolled members and account information for each

member, the account information including a current incentive point balance; identify a customer as an enrolled member; query the database to determine, in real time, the identified customer's incentive point balance; use the identified customer's incentive point balance to determine whether the customer is eligible for a discount on goods being purchased by the customer and, if so, offer a discount to the customer in exchange for redemption of all or part of the customer's incentive point balance; and, in response to the customer's directive, redeem all or part of the customer's incentive point balance to provide a purchase discount to the customer, update the database to reflect the redemption, and provide a receipt to the customer that details the redemption and the purchase discount.

[0035] Again, generally speaking, and without intending to be limiting, another aspect of the invention relates to article(s)-of-manufacture for operating an Internet-accessible incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to: provide, via an Internet-accessible server, a plurality of Web pages viewable by Internet-connected users using browser software; provide, through the Internet-accessible server, a plurality of services that allow the Internet-connected users to accrue incentive points and to utilize accrued incentive points; provide a Web homepage, having at least some of the following: (a) an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process; (b) an informational link, selectable by an Internet-connected user to provide information concerning the incentive marketing program; (c) a login link, selectable by an Internet-connected user to access user account information; (d) a redemption homepage link, selectable by an Internet-connected user to access a redemption homepage; and at least some of the following: (a) at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer; (b) at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable the user to enroll in a third-party program from which the user can accrue incentive points; (c) at least one direct-access shopping link, selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and (d) at least one direct-access redemption link, the direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit the user to redeem the depicted award; and provide, on the redemption homepage, at least some of the following: (a) a charitable donation link, selectable by an Internet-connected user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations; (b) an assignment link, selectable by an Internet-connected user to initiate a process whereby the user can assign incentive points from his/her account to another user's account; (c) a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services; (d) an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and (e) a plurality of direct-access redemption links, each including an associated depiction of an award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0036] Still further aspects of the invention relate to alternative combinations, or sub-combinations, of the above-recited elements and/or actions, consistent with, or in furtherance of, one or more objectives of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0037] Various aspects, features and advantages of the instant invention are depicted in the accompanying set figures, which is intended to be illustrative, rather than limiting, and in which:

[0038] **FIG. 1** depicts an illustrative embodiment of certain aspects of the present invention;

[0039] **FIG. 2** depicts an alternative illustrative embodiment of certain aspects of the present invention;

[0040] FIGS. **3A-B** collectively depict an illustrate embodiment of an Internet-accessible homepage in accordance with a preferred embodiment of the invention;

[0041] **FIG. 4** depicts an illustrative embodiment of an Internet-accessible redemption homepage in accordance with a preferred embodiment of the invention; and,

[0042] **FIG. 5** is a flowchart depicting an illustrative point-of-purchase redemption process in accordance with the present invention.

DESCRIPTION OF THE PRESENTLY-PREFERRED EMBODIMENT

[0043] Referring initially to **FIG. 1**, a computerized cash register **11** receives customer ID data from a customer identification device **10** and queries an in-store server **14** to identify a customer as an incentive program participant. If the customer is identified as a participant, in-store server **14** retrieves relevant account information, including a current incentive point balance, for use in processing incentive point transactions.

[0044] In-store server **14** may store (or cache) local account data itself, or may retrieve requested data, as needed, from a central server **16**. A computer network infrastructure **15** connects in-store server **14** and central server **16**. Network infrastructure **15** preferably comprises a part of the Internet, but can also be any sort of private or semi-private WAN or LAN.

[0045] A point-of-sale receipt printer **13** permits generation of a customer receipt **12** containing various account information. Among the information that receipt **12** might contain is:

[0046] the customer's current incentive point balance;

[0047] a record of a customer's point-of-purchase redemption transaction;

[0048] promotions concerning new or additional ways that the customer can accrue incentive points;

[0049] promotions related to new or additional ways that the customer can redeem or use incentive points;

[0050] pre-approved credit card offers.

[0051] Referring now to **FIG. 2**, an alternative embodiment of the system depicted in **FIG. 1** replaces computerized cash register **11** and in-store server **14** (which components are more typical of a large-store environment) with a single in-store computer **20**. In-store computer **20** may comprise a personal computer, a computerized cash register, or any sort of computational device having the requisite data input and network interfacing capabilities. Using computer **20**, one can perform all of the tasks necessary to operate an incentive program, in accordance with the present invention.

[0052] Referring now to FIGS. **3A-B**, which collectively depict an illustrate embodiment of an Internet-accessible homepage in accordance with a preferred ("www.greenpoints.com") embodiment of the invention, the depicted homepage includes:

[0053] an enrollment link (labeled "Enroll"), selectable by an Internet-connected user to initiate an on-line enrollment process;

[0054] a plurality of informational links (under heading "Here's how it works. . . "), selectable by an Internet-connected user to provide information concerning the incentive marketing program;

[0055] a login link (labeled "->Log in"), selectable by an Internet-connected user to access user account information;

[0056] at least one time-limited incentive link (labeled "Hertz is extending their Double Greenpoints offer until June 30th . . . "), selectable by an Internet-connected user to access a time-limited incentive offer;

[0057] a plurality of incentive-referral enrollment links (e.g., those labeled "Internet connection with EarthLink," and "Sprint Nickel Nights"), selectable by an Internet-connected user to direct the user to enroll in a third-party program from which the user can accrue incentive points;

[0058] a plurality of direct-access shopping links (e.g., those labeled "FTD.com,""Sparks.com,""Outpost.com," etc.), each selectable by an Internet-connected user to permit the user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor;

[0059] a plurality of direct-access redemption links (e.g., those labeled "Silver-Plated Tea Box,""Size 5 Tournament Soccer Ball," etc.), each including an associated depiction of an award and an incentive point total needed to redeem the award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award;

[0060] a brick-and-mortar customer referral link (labeled "Foodtown"), selectable by an Internet-connected user to permit the user to utilize incentive points earned in connection with brick-and-mortar purchases; and,

[0061] a primary redemption link (labeled "use greenpoints"), selectable by an Internet-connected user to navigate the user to a redemption homepage.

[0062] Referring now to **FIG. 4**, which depicts an illustrate embodiment of an Internet-accessible redemption homepage in accordance with a preferred ("www.greenpoints.com") embodiment of the invention, the depicted redemption homepage includes:

[0063] a charitable donation link (labeled "Make a donation"), selectable by an Internet-connected user to initiate a process whereby the user can donate incentive points from his/her account to one or more charitable organizations;

[0064] an assignment link (labeled "Make a friend smile"), selectable by an Internet-connected user to initiate a process whereby the user can assign incentive points from his/her account to another user's account;

[0065] a travel redemption link (labeled "turn greenpoints into Travel!"), selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

[0066] an awards catalog search tool (labeled "Search Greenpoints Reward Catalog"), engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii); and,

[0067] a plurality of direct-access redemption links (e.g., **5** those labeled "BLOCKBUSTER Movie Rental,""Soft-Touch Volleyball," etc.), each including an associated depiction of an award, the direct-access redemption links each selectable by an Internet-connected user to permit the user to redeem the depicted award.

[0068] Referring now to **FIG. 5**, which shows a flowchart depicting an illustrative point-of-purchase redemption process in accordance with the present invention, the process begins by identifying **30** a customer as an incentive program participant. This is preferably done by comparing data from a customer identification device (such as a bar code scanner that reads a card presented by the customer) to data stored in database **32**. Once the customer is identified as a participant, the number of available incentive points in the customer's account is determined **31**, preferably by querying database **32**.

[0069] The system then identifies **33** the items that the customer is presenting for purchase, and may also identify any incentive offers associated with the identified items. Next, all of the above information is processed to determine **34** (preferably in real time) the redemption options that may be available to the customer. Such redemption options may include, but are not limited to, discounts on certain items the customer is purchasing, cash rebates, movie passes, etc. The available redemption options are then presented **35** to the customer using, for example, a touch-screen display.

[0070] The customer then selects those redemption options that he/she wishes to exercise, and the system then executes **36** the selected redemptions. Finally, customer account data in database **32** is updated **37** to reflect the redemption(s), and a customer receipt is **39** is generated **38**.

[0071] Those skilled in the art will appreciate that the invention may be alternatively implemented, and/or improved/enhanced, through use of a variety of presently-existing technologies. Thus, for example, it is within the scope of the present invention to permit users to access the www.greenpoints.com site via not only conventional, PC-type computers, but also cellular and wireless devices, palm-top computers, cable television set-top boxes, in-store kiosks, and the like.

[0072] Thus, while the invention has been described by recitation of its various aspects/features and an illustrative embodiment thereof, those skilled in the art will recognize that alternative elements and techniques, and/or combinations and sub-combinations of the described elements and techniques, can be substituted for, or added to, those described herein. The present invention, therefore, should not be limited to, or defined by, the specific apparatus, methods, and articles-of-manufacture described herein, but rather by the appended claims, which are intended to be construed in accordance with well-settled principles of claim construction, including, but not limited to, the following:

[0073] Limitations should not be read from the specification or drawings into the claims (e.g., if the claim calls for a "chair," and the specification and drawings show a rocking chair, the claim term "chair" should not be limited to a rocking chair, but rather should be construed to cover any type of "chair").

[0074] The words "comprising,""including," and "having" are always open-ended, irrespective of whether they appear as the primary transitional phrase of a claim, or as a transitional phrase within an element or sub-element of the claim (e.g., the claim "a widget comprising: A; B; and C" would be infringed by a device containing 2A's, B, and 3C's; also, the claim "a gizmo comprising: A; B, including X, Y, and Z; and C, having P and Q" would be infringed by a device containing 3A's, 2X's, 3Y's, Z, 6P's, and Q).

[0075] The indefinite articles "a" or "an" mean "one or more";

[0076] where, instead, a purely singular meaning is intended, a phrase such as "one,""only one," or "a single," will appear.

[0077] Where the phrase "means for" precedes a data processing or manipulation "function," it is intended that the resulting means-plus-function element be construed to cover any, and all, computer implementation(s) of the recited "function" using any standard programming techniques known by, or available to, persons skilled in the computer programming arts.

[0078] A claim that contains more than one computer-implemented means-plus-function element should not be construed to require that each means-plus-function element must be a structurally distinct entity (such as a particular piece of hardware or block of code); rather, such claim should be construed merely to require that the overall combination of hardware/firmware/software which implements the invention must, as a whole, implement at least the function(s) called for by the claims.

What is claimed is:

1. A method for operating an Internet-accessible incentive marketing program, comprising:

providing, via an Internet-accessible server, a plurality of html pages, including a homepage, said html pages viewable by Internet-connected users using browser software;

providing, through said Internet-accessible server, a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points;

providing, on said homepage, at least the following:

an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process;

an informational link, selectable by an Internet-connected user to provide information concerning said incentive marketing program;

a login link, selectable by an Internet-connected user to access user account information; and,

at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

2. A method for operating an Internet-accessible incentive marketing program, as defined in claim 1, wherein the third-party program associated with the at least one incentive-referral enrollment link relates to Internet access services.

3. A method for operating an Internet-accessible incentive marketing program, as defined in claim 1, wherein the third-party program associated with the at least one incentive-referral enrollment link relates to long-distance telephone services.

4. A method for operating an Internet-accessible incentive marketing program, as defined in claim 1, wherein the third-party program associated with the at least one incentive-referral enrollment link relates to on-line auction services.

5. A method for operating an Internet-accessible incentive marketing program, comprising:

providing, via an Internet-accessible server, a plurality of html pages, including a homepage, said html pages viewable by Internet-connected users using browser software;

providing, through said Internet-accessible server, a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points; and,

providing, on said homepage, a brick-and-mortar customer referral link, selectable by an Internet-connected user to permit said user to utilize incentive points earned in connection with brick-and-mortar purchases.

6. A method for operating an Internet-accessible incentive marketing program, as defined in claim 5, further including providing, on said homepage, at least one of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and/or,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

7. A method for operating an Internet-accessible incentive marketing program, as defined in claim 5, further including providing, on said homepage, at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

8. A method for operating an Internet-accessible incentive marketing program, as defined in claim 5, further including providing, on said homepage, at least three of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

9. A method for operating an Internet-accessible incentive point redemption system, comprising:

maintaining, on an Internet-accessible server, data corresponding to a multiplicity of customer accounts;

providing an Internet-accessible homepage through which an Internet-connected user can access said server to interrogate or utilize his/her account;

providing, on said Internet-accessible homepage, a primary redemption link, selectable by an Internet-connected user to navigate said user to a redemption homepage;

providing, on said redemption homepage, at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii);

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

10. A method for operating an Internet-accessible incentive point redemption system, as defined in claim 9, wherein said redemption homepage provides at least three of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii);

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

11. A method for operating an Internet-accessible incentive point redemption system, as defined in claim 9, wherein said redemption homepage provides at least four of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii);

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

12. A method for operating an Internet-accessible incentive marketing program, comprising:

providing, via an Internet-accessible server, a plurality of Web pages viewable by Internet-connected users using browser software;

providing, through said Internet-accessible server, a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points;

providing a Web homepage, having at least the following:

an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process;

an informational link, selectable by an Internet-connected user to provide information concerning said incentive marketing program;

a login link, selectable by an Internet-connected user to access user account information;

a redemption homepage link, selectable by an Internet-connected user to access a redemption homepage; and,

at least one of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award; and,

providing, on said redemption homepage, at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

13. A method for operating an Internet-accessible incentive marketing program, as defined in claim **12**:

wherein said Web homepage provides at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated

depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award;

and wherein said redemption homepage provides at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

14. A method for operating an Internet-accessible incentive marketing program, as defined in claim **12**:

wherein said Web homepage provides at least three of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an

Internet-connected user to permit said user to redeem the depicted award;

and wherein said redemption homepage provides at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

15. A method for operating an Internet-accessible incentive marketing program, as defined in claim **12**:

wherein said Web homepage provides at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award;

and wherein said redemption homepage provides at least three of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

16. A method for operating an Internet-accessible incentive marketing program, as defined in claim **12**:

wherein said Web homepage provides at least three of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award;

and wherein said redemption homepage provides at least three of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

17. A method for operating an incentive marketing program, comprising:

maintaining, in a database, a list of enrolled members and account information for each member, said account information including a current incentive point balance;

identifying a customer as an enrolled member;

using the database to determine, in real time, the identified customer's incentive point balance;

using the identified customer's incentive point balance to determine whether the customer is eligible for a discount on goods being purchased by the customer and, if so, offering a discount to the customer in exchange for redemption of all or part of the customer's incentive point balance; and,

in response to the customer's directive, redeeming all or part of the customer's incentive point balance to provide a purchase discount to the customer, updating the database to reflect the redemption, and providing a receipt to the customer that details the redemption and the purchase discount.

18. A method for operating an incentive marketing program, as defined in claim 17, wherein identifying a customer as an enrolled member comprises comparing data received from a magnetic stripe reader at the point-of-purchase to data stored in the database.

19. A method for operating an incentive marketing program, as defined in claim 17, wherein identifying a customer as an enrolled member comprises comparing data received from an optical bar code reader at the point-of-purchase to data stored in the database.

20. A method for operating an incentive marketing program, as defined in claim 17, wherein identifying a customer as an enrolled member comprises comparing data received from a smart card reader at the point-of-purchase to data stored in the database.

21. A method for operating an incentive marketing program, as defined in claim 17, wherein identifying a customer as an enrolled member comprises comparing data entered into a data entry device at the point-of-purchase to data stored in the database.

22. A method for operating an incentive marketing program, comprising:

maintaining, in a database, a list of enrolled members and account information for each member, said account information including a current incentive point balance;

identifying a customer, at a point-of-purchase location, as an enrolled member; using the database to determine, in real time, the identified customer's incentive point balance;

determining whether any items that the identified customer has presented for purchase can be acquired or discounted through redemption of incentive point's in the customer's account and, if so, offering to redeem incentive points in exchange for discount(s) on, and/or acquisition(s)of, one or more items that the customer has presented for purchase.

23. A method of operating a universal incentive marketing program that permits registered users to accrue incentive points for both on-line and in-store transactions, comprising:

maintaining a network-accessible central server that includes a database of account information for registered users;

providing a customer identification device at a retail point-of-purchase location that permits identification of a customer as a registered user;

adding incentive points to the accounts of customers identified as registered users for in-store purchases made by said users;

providing an Internet-accessible homepage linked to the central server to permit identification of an Internet-connected user as a registered user, the homepage further including at least two of the following:

at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by the user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by the user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

24. A method of operating a universal incentive marketing program, as defined in claim 23, wherein the homepage includes at least three of the following:

at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by the user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by the user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

25. A method of operating a universal incentive marketing program, as defined in claim 23, wherein the homepage includes:

at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer; at least one incentive-referral enrollment link, selectable by the user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by the user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable the user to permit said user to redeem the depicted award.

26. A method of operating a universal incentive marketing program, as defined in claim 23, wherein the homepage includes a link to a redemption homepage, and the redemption homepage includes at least two of the following:

a charitable donation link, selectable by the user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by the user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by the user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by the user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

27. A method of operating a universal incentive marketing program, as defined in claim 23, wherein the homepage includes a link to a redemption homepage, and the redemption homepage includes at least three of the following:

a charitable donation link, selectable by the user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by the user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by the user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by the user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

28. A method of operating a universal incentive marketing program, as defined in claim 23, wherein the homepage includes a link to a redemption homepage, and the redemption homepage includes at least two of the following:

a charitable donation link, selectable by the user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by the user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by the user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by the user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

29. An Internet-accessible incentive marketing system, comprising:

an Internet-accessible server, configured to serve a plurality of html pages, including a homepage, viewable by Internet-connected users using browser software;

said server further configured to provide a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points;

said homepage including a brick-and-mortar customer referral link, selectable by an Internet-connected user to permit said user to utilize incentive points earned in connection with brick-and-mortar purchases.

30. An Internet-accessible incentive marketing system, as defined in claim 29, wherein said homepage further includes at least one of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and/or,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

31. An Internet-accessible incentive marketing system, as defined in claim 29, wherein said homepage further includes at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

32. An Internet-accessible incentive marketing system, as defined in claim 29, wherein said homepage further includes at least three of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

33. A universal incentive marketing system that permits registered users to accrue incentive points for both on-line and in-store transactions, comprising:

a network-accessible central server that includes a database of account information for registered users;

a customer identification device, comprising one of a bar code reader, a magnetic stripe reader, a smart card reader, or a manual data entry device, the customer identification device permitting identification of a customer as a registered user;

an incentive point accounting module, configured to add incentive points to the accounts of customers identified as registered users for in-store purchases made by said users; and,

an Internet-accessible homepage linked to the central server to permit identification of an Internet-connected user as a registered user, the homepage further including at least two of the following:

at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by the user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by the user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

34. A universal incentive marketing system, as defined in claim 33, wherein the homepage includes at least three of the following:

at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by the user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by the user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

35. A universal incentive marketing system, as defined in claim 33, wherein the homepage includes:

at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by the user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by the user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable the user to permit said user to redeem the depicted award.

36. A universal incentive marketing system, as defined in claim 33, wherein the homepage includes a link to a redemption homepage, and the redemption homepage includes at least two of the following:

a charitable donation link, selectable by the user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by the user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by the user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by the user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

37. A universal incentive marketing system, as defined in claim 33, wherein the homepage includes a link to a redemption homepage, and the redemption homepage includes at least three of the following:

a charitable donation link, selectable by the user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by the user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by the user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by the user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

38. A method of operating a universal incentive marketing system, as defined in claim 33, wherein the homepage includes a link to a redemption homepage, and the redemption homepage includes at least two of the following:

a charitable donation link, selectable by the user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by the user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by the user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by the user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

39. An Internet-accessible incentive marketing system, comprising:

an Internet-accessible server, configured to serve up a plurality of html pages, including a homepage, said html pages viewable by Internet-connected users using browser software;

said Internet-accessible server further configured to serve up a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points;

said homepage including at least the following:

an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process;

an informational link, selectable by an Internet5 connected user to provide information concerning said incentive marketing program;

a login link, selectable by an Internet-connected user to access user account information; and,

at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said

user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

40. An Internet-accessible incentive marketing system, as defined in claim 39, wherein the third-party program associated with the at least one incentive-referral enrollment link relates to Internet access services.

41. An Internet-accessible incentive marketing system, as defined in claim 39, wherein the third-party program associated with the at least one incentive-referral enrollment link relates to long-distance telephone services.

42. An Internet-accessible incentive marketing system, as defined in claim 39, wherein the third-party program associated with the at least one incentive-referral enrollment link relates to on-line auction services.

43. A method for operating an incentive marketing system, comprising:

a database containing a list of enrolled members and account information for each member, said account information including a current incentive point balance;

a customer identification device, located at a point-of-purchase, and configured to identify a customer as an enrolled member;

a customer information query module, configured to query said database to determine, in real time, the identified customer's incentive point balance; and,

a point-of-purchase redemption processing module, configured to determine, in real time, whether any items that the identified customer has presented for purchase can be acquired or discounted through redemption of incentive point's in the customer's account and, if so, to present an offer to redeem incentive points in exchange for discount(s) on, and/or acquisition(s)of, one or more items that the customer has presented for purchase.

44. An Internet-accessible incentive point redemption system, comprising:

an Internet-accessible server that stores data corresponding to a multiplicity of customer accounts;

an Internet-accessible homepage through which an Internet-connected user can access said server to interrogate or utilize his/her account;

said Internet-accessible homepage including a primary redemption link, selectable by an Internet-connected user to navigate said user to a redemption homepage;

said redemption homepage including at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii);

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

45. An Internet-accessible incentive point redemption system, as defined in claim 44, wherein said redemption homepage provides at least three of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii);

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

46. An Internet-accessible incentive point redemption system, as defined in claim 44, wherein said redemption homepage provides at least four of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii);

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

47. An incentive marketing system, comprising:

a database that includes a list of enrolled members and account information for each member, said account information including a current incentive point balance;

means for identifying a customer as an enrolled member;

means for using the database to determine, in real time, the identified customer's incentive point balance;

means for using the identified customer's incentive point balance to determine whether the customer is eligible for a discount on goods being purchased by the customer and, if so, offering a discount to the customer in exchange for redemption of all or part of the customer's incentive point balance; and,

means, responsive to the customer's directive, for redeeming all or part of the customer's incentive point balance to provide a purchase discount to the customer, updating the database to reflect the redemption, and providing a receipt to the customer that details the redemption and the purchase discount.

48. An incentive marketing system, as defined in claim 47, wherein the means for identifying a customer as an enrolled member comprises means for comparing data received from a magnetic stripe reader at the point-of-purchase to data stored in the database.

49. An incentive marketing system, as defined in claim 47, wherein the means for identifying a customer as an enrolled member comprises means for comparing data received from an optical bar code reader at the point-of-purchase to data stored in the database.

50. An incentive marketing system, as defined in claim 47, wherein the means for identifying a customer as an enrolled member comprises means for comparing data received from a smart card reader at the point-of-purchase to data stored in the database.

51. An incentive marketing system, as defined in claim 47, wherein the means for identifying a customer as an enrolled member comprises means for comparing data entered into a data entry device at the point-of-purchase to data stored in the database.

52. An Internet-accessible incentive marketing system, comprising:

an Internet-accessible server, configured to serve up a plurality of Web pages viewable by Internet-connected users using browser software;

said Internet-accessible server further configured to serve up a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points;

a Web homepage, having at least the following:

an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process;

an informational link, selectable by an Internet-connected user to provide information concerning said incentive marketing program;

a login link, selectable by an Internet-connected user to access user account information;

a redemption homepage link, selectable by an Internet-connected user to access a redemption homepage; and,

at least one of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award; and,

a redemption homepage that includes at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

53. An Internet-accessible incentive marketing system, as defined in claim **52**:

wherein said Web homepage includes at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award;

and wherein said redemption homepage includes at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

54. An Internet-accessible incentive marketing system, as defined in claim **52**:

wherein said Web homepage includes at least three of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award;

and wherein said redemption homepage includes at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

55. An Internet-accessible incentive marketing system, as defined in claim **52**:

wherein said Web homepage includes at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award;

and wherein said redemption homepage includes at least three of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

56. A method for operating an Internet-accessible incentive marketing system, as defined in claim **52**:

wherein said Web homepage includes at least three of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award;

and wherein said redemption homepage includes at least three of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

57. Article(s)-of-manufacture for operating an incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to:

maintain, in a database, a list of enrolled members and account information for each member, said account information including a current incentive point balance;

identify a customer, at a point-of-purchase location, as an enrolled member;

use the database to determine, in real time, the identified customer's incentive point balance;

determine, in real time, whether any items that the identified customer has presented for purchase can be acquired or discounted through redemption of incentive point's in the customer's account and, if so, offer to redeem incentive points in exchange for discount(s) on, and/or acquisition(s) of, one or more items that the customer has presented for purchase.

58. Article(s)-of-manufacture for operating an Internet-accessible incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to:

provide, via an Internet-accessible server, a plurality of html pages, including a homepage, said html pages viewable by Internet-connected users using browser software;

provide, through said Internet-accessible server, a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points;

provide, on said homepage, at least the following:

an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process;

an informational link, selectable by an Internet-connected user to provide information concerning said incentive marketing program;

a login link, selectable by an Internet-connected user to access user account information; and,

at least three of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

59. Article(s)-of-manufacture for operating a universal incentive marketing program that permits registered users to accrue incentive points for both on-line and in-store trans-

actions, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to:

maintain a network-accessible central server that includes a database of account information for registered users;

receive data from a customer identification device at a retail point-of-purchase location, so as to permit identification of a customer as a registered user;

add incentive points to the accounts of customers identified as registered users for in-store purchases made by said users; and,

provide an Internet-accessible homepage linked to the central server to permit identification of an Internet-connected user as a registered user, the homepage further including at least two of the following:

at least one time-limited incentive link, selectable by the user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by the user to direct said user to enroll in a third-party program from which said user can accrue incentive points;

a plurality of direct-access shopping links, each selectable by the user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points therefor; and,

a plurality of direct-access redemption links, each including an associated depiction of an award and an incentive point total needed to redeem said award, said direct-access redemption links each selectable by the user to permit said user to redeem the depicted award.

60. Article(s)-of-manufacture for operating an Internet-accessible incentive point redemption system, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to:

maintain, on an Internet-accessible server, data corresponding to a multiplicity of customer accounts;

provide an Internet-accessible homepage through which an Internet-connected user can access said server to interrogate or utilize his/her account;

provide, on said Internet-accessible homepage, a primary redemption link, selectable by an Internet-connected user to navigate said user to a redemption homepage;

provide, on said redemption homepage, at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search an awards catalog by: (i) points required for redemption, (ii) category of reward, or both (i) and (ii); and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

61. Article(s)-of-manufacture for operating an Internet-accessible incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to:

provide, via an Internet-accessible server, a plurality of html pages, including a homepage, said html pages viewable by Internet-connected users using browser software;

provide, through said Internet-accessible server, a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points; and,

provide, on said homepage, a brick-and-mortar customer referral link, selectable by an Internet-connected user to permit said user to utilize incentive points earned in connection with brick-and-mortar purchases.

62. Article(s)-of-manufacture for operating an incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to:

maintain, in a database, a list of enrolled members and account information for each member, said account information including a current incentive point balance;

identify a customer as an enrolled member;

query the database to determine, in real time, the identified customer's incentive point balance;

use the identified customer's incentive point balance to determine whether the customer is eligible for a discount on goods being purchased by the customer and, if so, offer a discount to the customer in exchange for redemption of all or part of the customer's incentive point balance; and,

in response to the customer's directive, redeem all or part of the customer's incentive point balance to provide a purchase discount to the customer, update the database to reflect the redemption, and provide a receipt to the customer that details the redemption and the purchase discount.

63. Article(s)-of-manufacture for operating an Internet-accessible incentive marketing program, the article(s)-of-manufacture comprising computer-readable media containing code which, when executed, causes a computer to:

provide, via an Internet-accessible server, a plurality of Web pages viewable by Internet-connected users using browser software;

provide, through said Internet-accessible server, a plurality of services that allow said Internet-connected users to accrue incentive points and to utilize accrued incentive points;

provide a Web homepage, having at least the following:

an enrollment link, selectable by an Internet-connected user to initiate an on-line enrollment process;

an informational link, selectable by an Internet-connected user to provide information concerning said incentive marketing program;

a login link, selectable by an Internet-connected user to access user account information;

a redemption homepage link, selectable by an Internet-connected user to access a redemption homepage; and,

at least two of the following:

at least one time-limited incentive link, selectable by an Internet-connected user to access a time-limited incentive offer;

at least one incentive-referral enrollment link, selectable by an Internet-connected user to enable said user to enroll in a third-party program from which said user can accrue incentive points;

at least one direct-access shopping link, selectable by an Internet-connected user to permit said user to transact on-line business with an associated merchant or service provider, and accrue incentive points based on the value of the transacted business; and,

at least one direct-access redemption link, said direct-access redemption link including an associated depiction of an award and being selectable by an Internet-connected user to permit said user to redeem the depicted award; and,

provide, on said redemption homepage, at least two of the following:

a charitable donation link, selectable by an Internet-connected user to initiate a process whereby said user can donate incentive points from his/her account to one or more charitable organizations;

an assignment link, selectable by an Internet-connected user to initiate a process whereby said user can assign incentive points from his/her account to another user's account;

a travel redemption link, selectable by an Internet-connected user to initiate a process whereby incentive points can be redeemed for travel services;

an awards catalog search tool, engageable by an Internet-connected user to search a catalog of awards that can be obtained using incentive points; and,

a plurality of direct-access redemption links, each including an associated depiction of an award, said direct-access redemption links each selectable by an Internet-connected user to permit said user to redeem the depicted award.

64. A method for operating an incentive marketing program, comprising:

maintaining, in a database, a list of enrolled members of said incentive marketing program;

identifying, in real time, at a point-of-purchase location, a customer as an enrolled member;

determining, in real time, whether said customer is eligible to participate in a promotion and, if so, presenting the promotion to the customer.

65. A method for operating an incentive marketing program, as defined in claim **64**, wherein said promotion comprises one or more of:

a discount on product(s) that the customer is purchasing;

a discount on product(s) that the customer is purchasing, contingent upon the customer purchasing selected additional product(s);

a pre-approved credit card offer;

a discount travel offer;

an offer to earn incentive points at an accelerated rate; and/or,

an offer to earn free merchandise if the customer makes additional purchases, satisfies certain conditions, or performs certain actions.

* * * * *

US 20030041045A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2003/0041045 A1

Sun (43) Pub. Date: Feb. 27, 2003

(54) METHOD OF IMPLEMENTING ELECTRONIC NAME CARD AND SYSTEM THEREOF

(76) Inventor: Way Sun, Shanghai (CN)

Correspondence Address:
ALSTON & BIRD LLP
BANK OF AMERICA PLAZA
101 SOUTH TRYON STREET, SUITE 4000
CHARLOTTE, NC 28280-4000 (US)

(21) Appl. No.: 09/738,479

(22) Filed: Dec. 15, 2000

(30) Foreign Application Priority Data

Jul. 5, 2000 (CN) 00109767.9

Publication Classification

(51) Int. Cl.[7] G06F 17/30; G06K 19/06

(52) U.S. Cl. 707/1; 707/104.1; 707/9; 235/492

(57) ABSTRACT

This invention discloses a method and its system for implementing electronic name card. With database in telecommunication, the method establishes virtual electronic name card, which is a summation of user's personal information marked by user's ID. The system is composed of a central transaction-processing server and several local transaction processing servers. The central transaction-processing server links with central database system, and the local transaction-processing server with local database system. The central transaction processing server connects with the local transaction processing server through the computer network, while the public telecommunication network is connected with the ENC system through the local transaction processing server. A user can dial a special access number to request the electronic name card services including query, modification, exchange, and fast call.

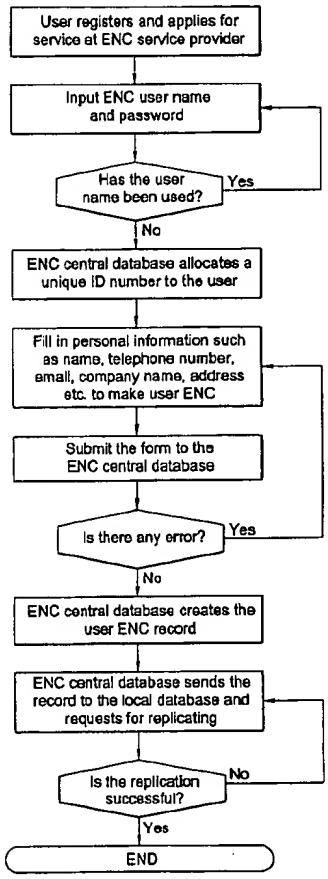


FIG. 1.

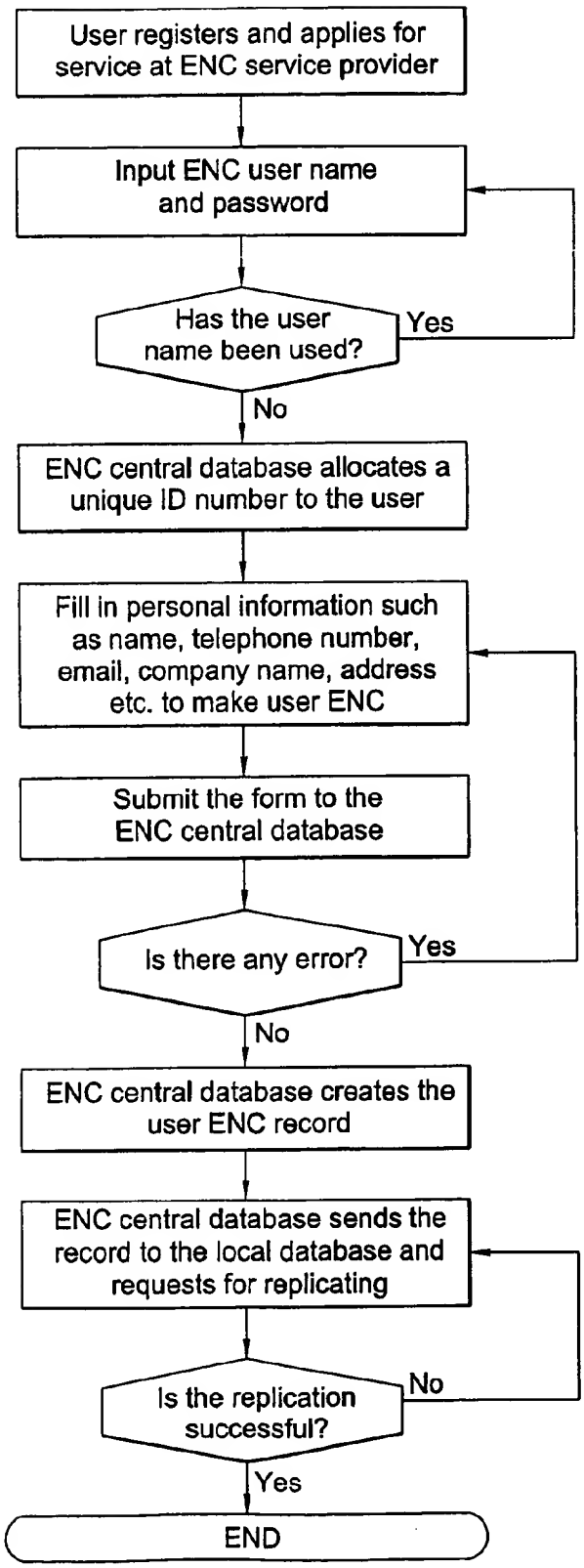
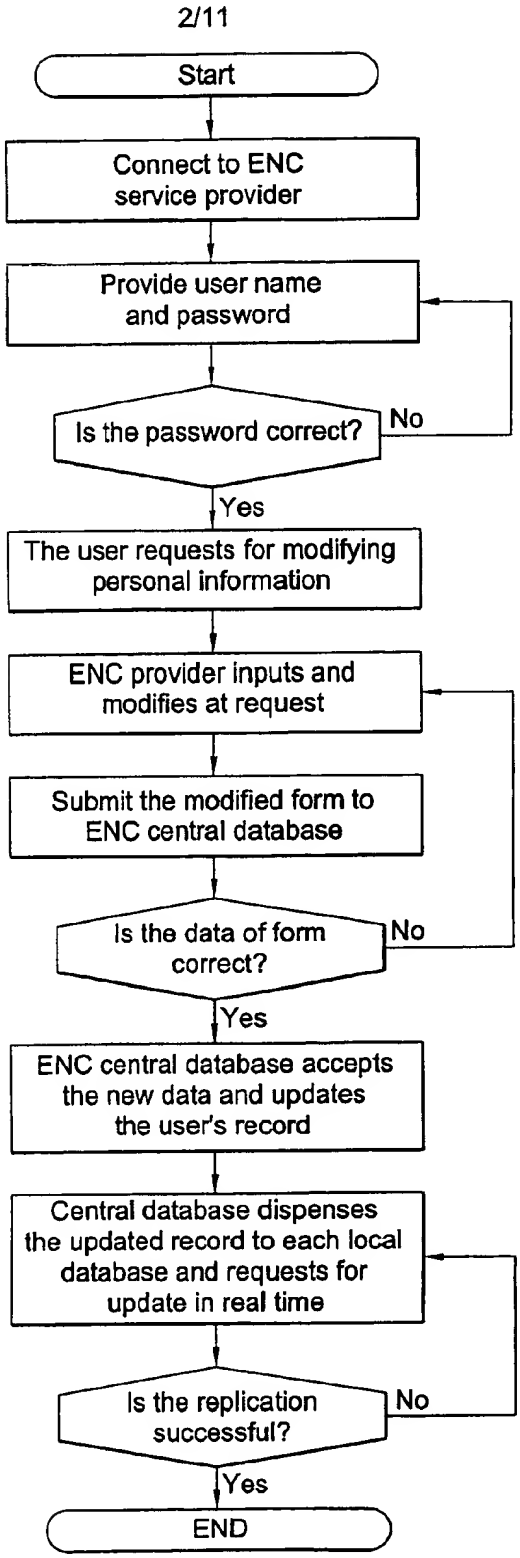

User registers and applies for service at ENC service provider
Input ENC user name and password
Has the user name been used?
Yes
No
ENC central database allocates a unique ID number to the user
Fill in personal information such as name, telephone number, email, company name, address etc. to make user ENC
Submit the form to the ENC central database
Is there any error?
Yes
No
ENC central database creates the user ENC record
ENC central database sends the record to the local database and requests for replicating
Is the replication successful?
No
Yes
END

FIG. 2.


2/11
Start
Connect to ENC service provider
Provide user name and password
Is the password correct?
No
Yes
The user requests for modifying personal information
ENC provider inputs and modifies at request
Submit the modified form to ENC central database
Is the data of form correct?
No
Yes
ENC central database accepts the new data and updates the user's record
Central database dispenses the updated record to each local database and requests for update in real time
Is the replication successful?
No
Yes
END

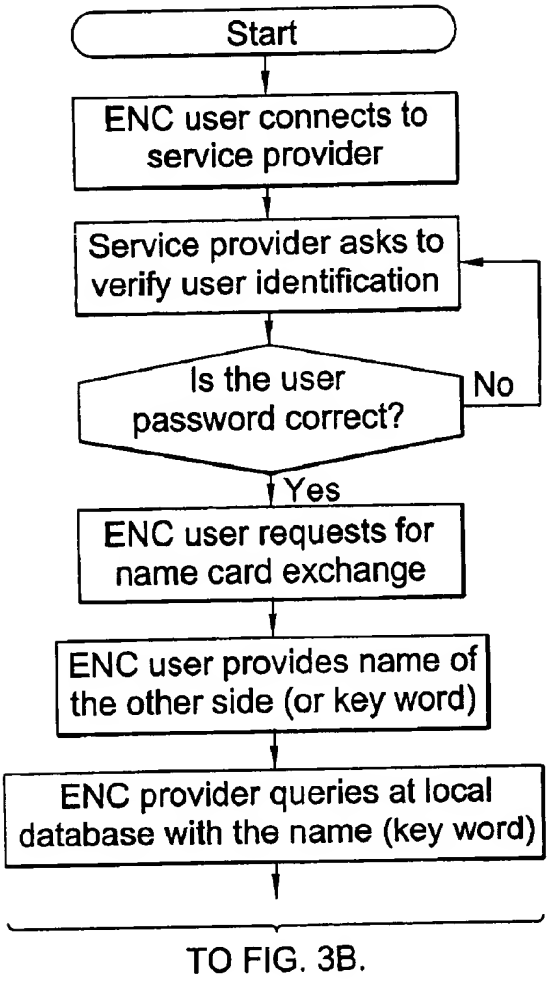

Start
ENC user connects to service provider
Service provider asks to verify user identification
Is the user password correct?
No
Yes
ENC user requests for name card exchange
ENC user provides name of the other side (or key word)
ENC provider queries at local database with the name (key word)
TO FIG. 3B.

FIG. 3A.

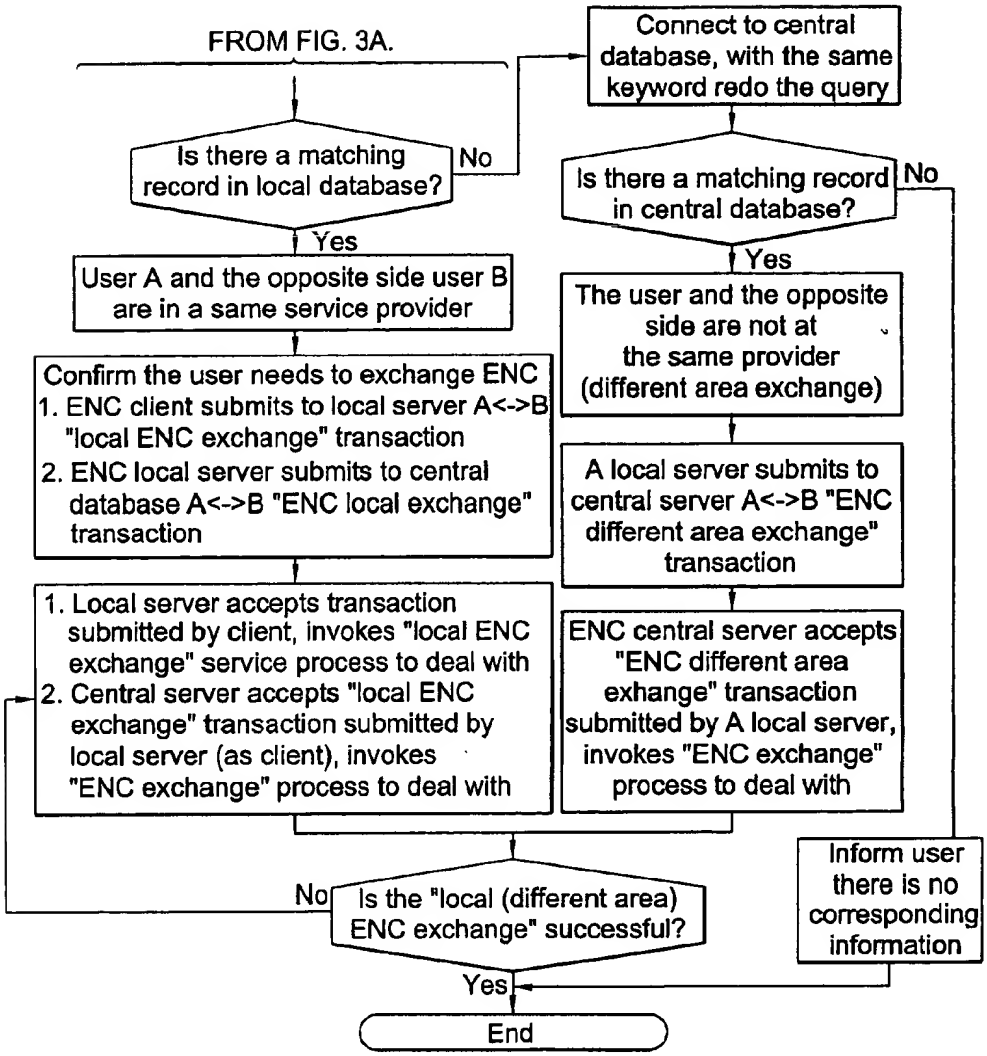

FROM FIG. 3A.
Is there a matching record in local database?
No
Yes
User A and the opposite side user B are in a same service provider
Confirm the user needs to exchange ENC
1. ENC client submits to local server A<->B "local ENC exchange" transaction
2. ENC local server submits to central database A<->B "ENC local exchange" transaction
1. Local server accepts transaction submitted by client, invokes "local ENC exchange" service process to deal with
2. Central server accepts "local ENC exchange" transaction submitted by local server (as client), invokes "ENC exchange" process to deal with
Connect to central database, with the same keyword redo the query
Is there a matching record in central database?
No
Yes
The user and the opposite side are not at the same provider (different area exchange)
A local server submits to central server A<->B "ENC different area exchange" transaction
ENC central server accepts "ENC different area exhange" transaction submitted by A local server, invokes "ENC exchange" process to deal with
Inform user there is no corresponding information
No
Is the "local (different area) ENC exchange" successful?
Yes
End

FIG. 3B.

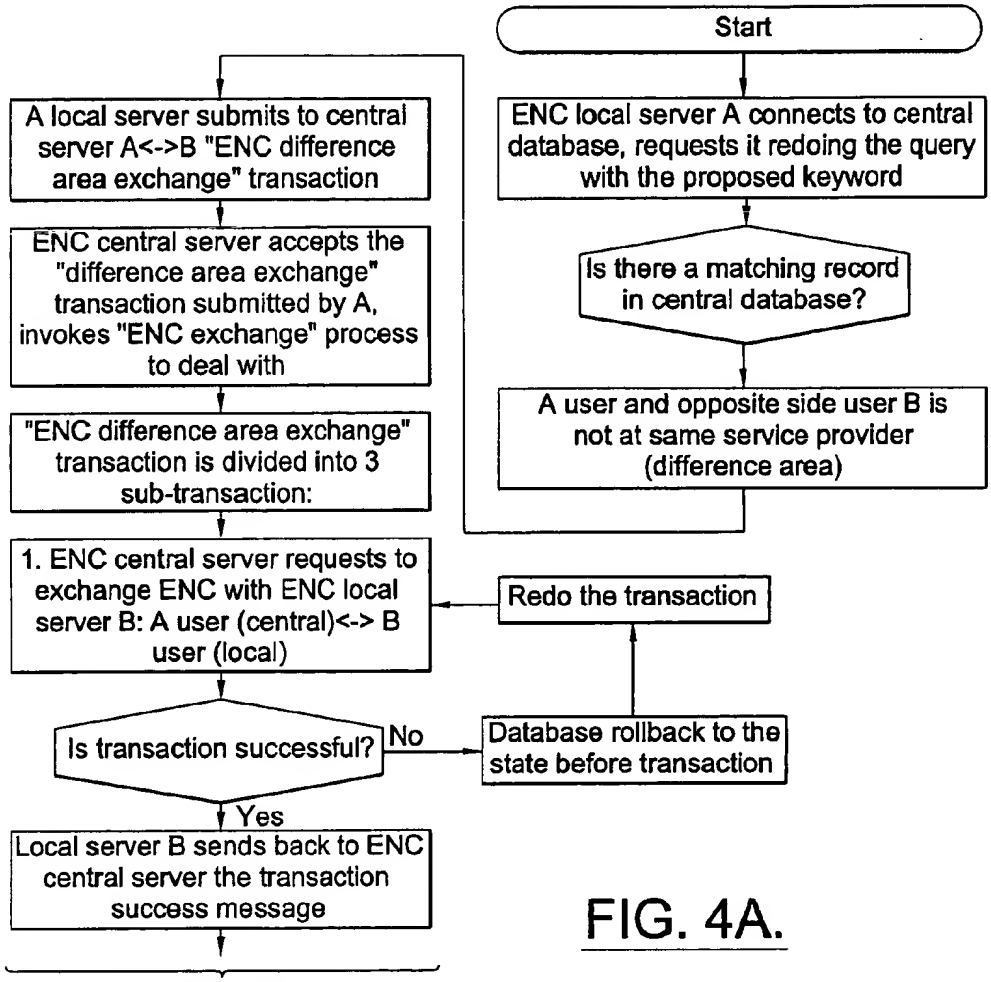

Start
ENC local server A connects to central database, requests it redoing the query with the proposed keyword
Is there a matching record in central database?
A user and opposite side user B is not at same service provider (difference area)
Redo the transaction
Database rollback to the state before transaction
No
Is transaction successful?
Yes
A local server submits to central server A<->B "ENC difference area exchange" transaction
ENC central server accepts the "difference area exchange" transaction submitted by A, invokes "ENC exchange" process to deal with
"ENC difference area exchange" transaction is divided into 3 sub-transaction:
1. ENC central server requests to exchange ENC with ENC local server B: A user (central)<-> B user (local)
Local server B sends back to ENC central server the transaction success message
TO FIG. 4B.

FIG. 4A.

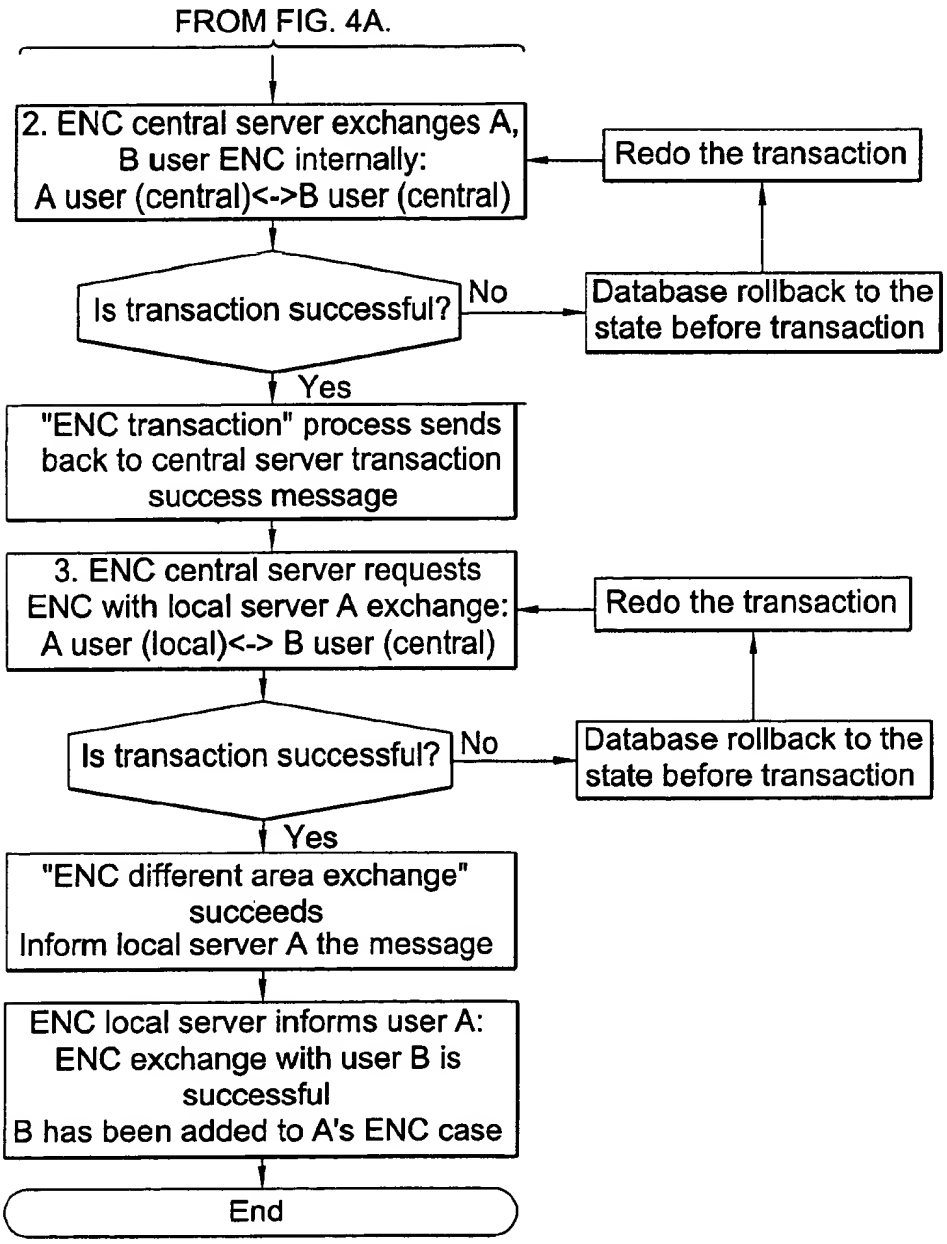

FROM FIG. 4A.
2. ENC central server exchanges A, B user ENC internally: A user (central)<->B user (central)
Redo the transaction
Is transaction successful?
No
Database rollback to the state before transaction
Yes
"ENC transaction" process sends back to central server transaction success message
3. ENC central server requests ENC with local server A exchange: A user (local)<-> B user (central)
Redo the transaction
Is transaction successful?
No
Database rollback to the state before transaction
Yes
"ENC different area exchange" succeeds Inform local server A the message
ENC local server informs user A: ENC exchange with user B is successful B has been added to A's ENC case
End

FIG. 4B.

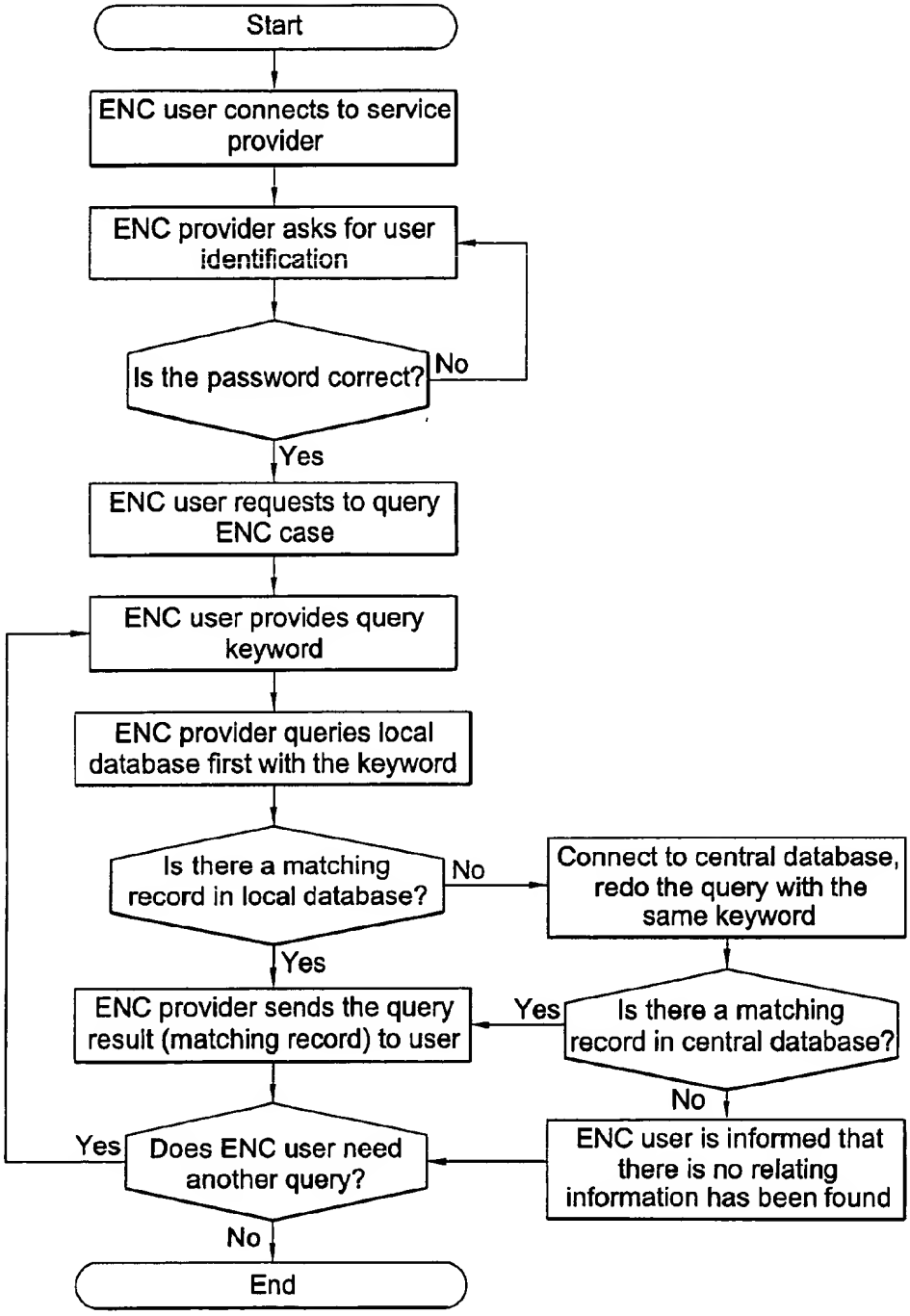

Start
ENC user connects to service provider
ENC provider asks for user identification
Is the password correct?
No
Yes
ENC user requests to query ENC case
ENC user provides query keyword
ENC provider queries local database first with the keyword
Is there a matching record in local database?
No
Connect to central database, redo the query with the same keyword
Yes
ENC provider sends the query result (matching record) to user
Yes
Is there a matching record in central database?
No
Yes
Does ENC user need another query?
ENC user is informed that there is no relating information has been found
No
End

FIG. 5.

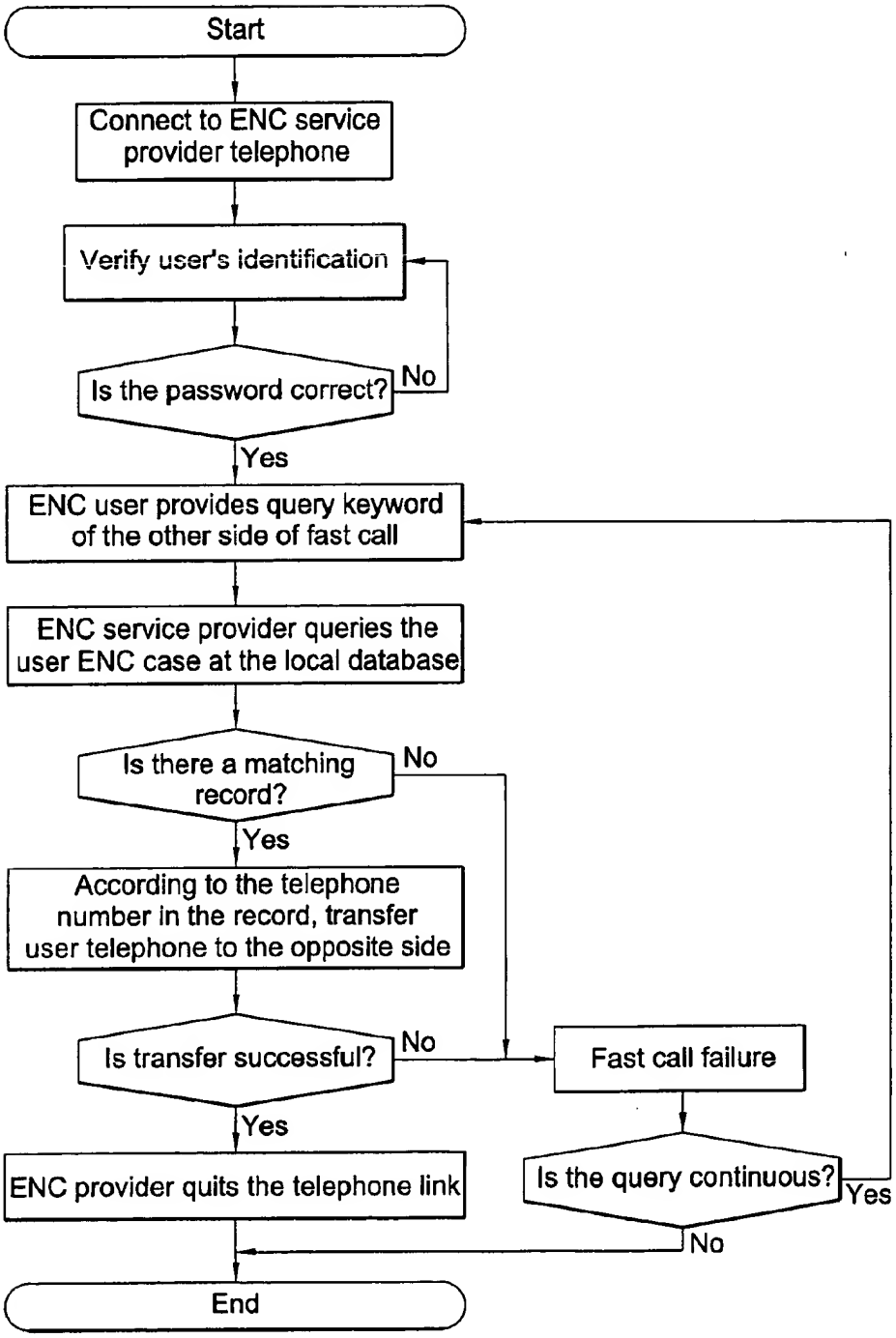

Start
Connect to ENC service provider telephone
Verify user's identification
Is the password correct?
No
Yes
ENC user provides query keyword of the other side of fast call
ENC service provider queries the user ENC case at the local database
Is there a matching record?
No
Yes
According to the telephone number in the record, transfer user telephone to the opposite side
Is transfer successful?
No
Fast call failure
Yes
ENC provider quits the telephone link
Is the query continuous?
Yes
No
End

FIG. 6.

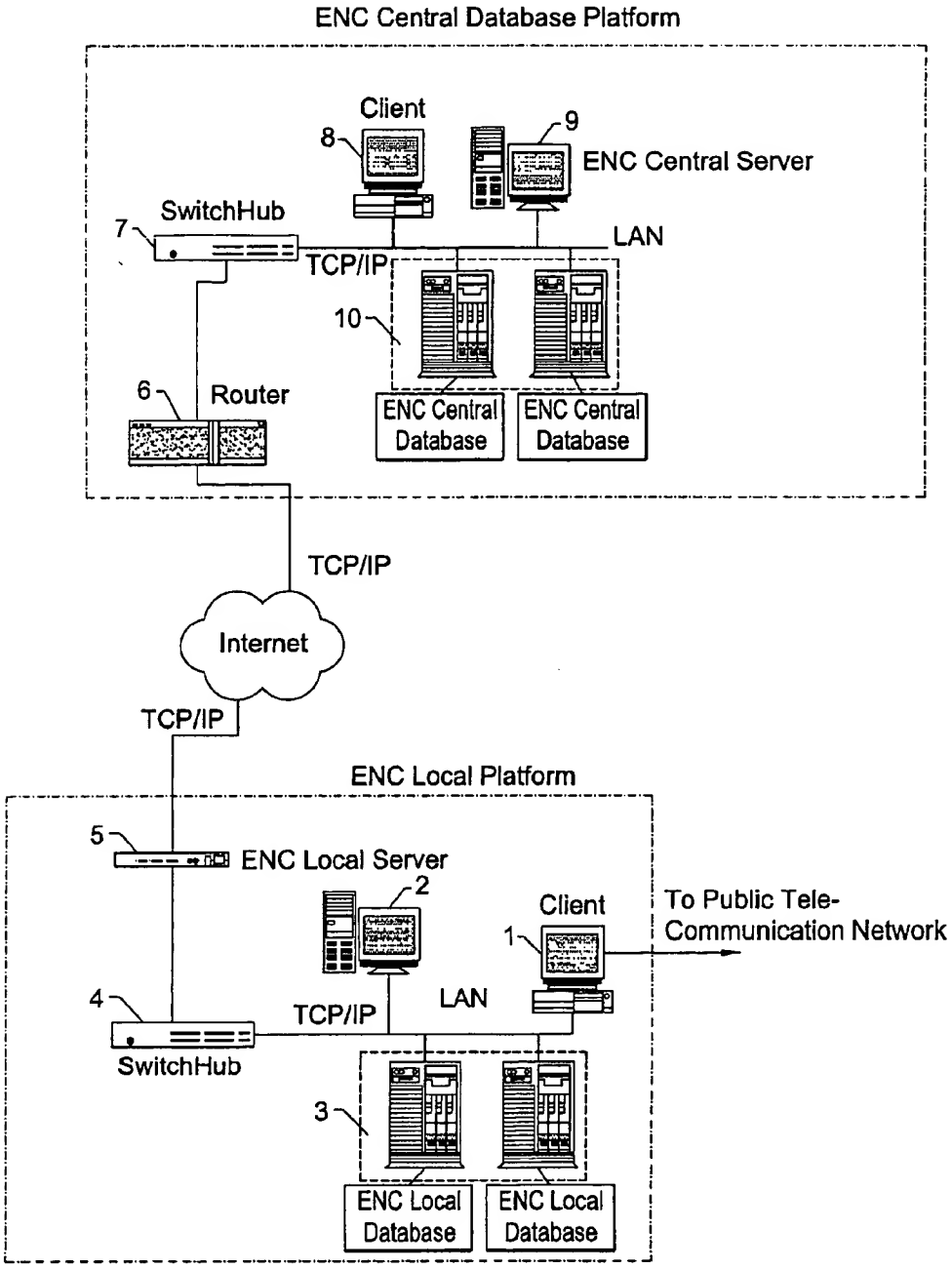

ENC Central Database Platform
Client
8
9
ENC Central Server
SwitchHub
7
LAN
TCP/IP
10
6
Router
ENC Central Database
ENC Central Database
TCP/IP
Internet
TCP/IP
ENC Local Platform
5
ENC Local Server
2
Client
1
To Public Tele-Communication Network
4
TCP/IP
LAN
SwitchHub
3
ENC Local Database
ENC Local Database

FIG. 7.

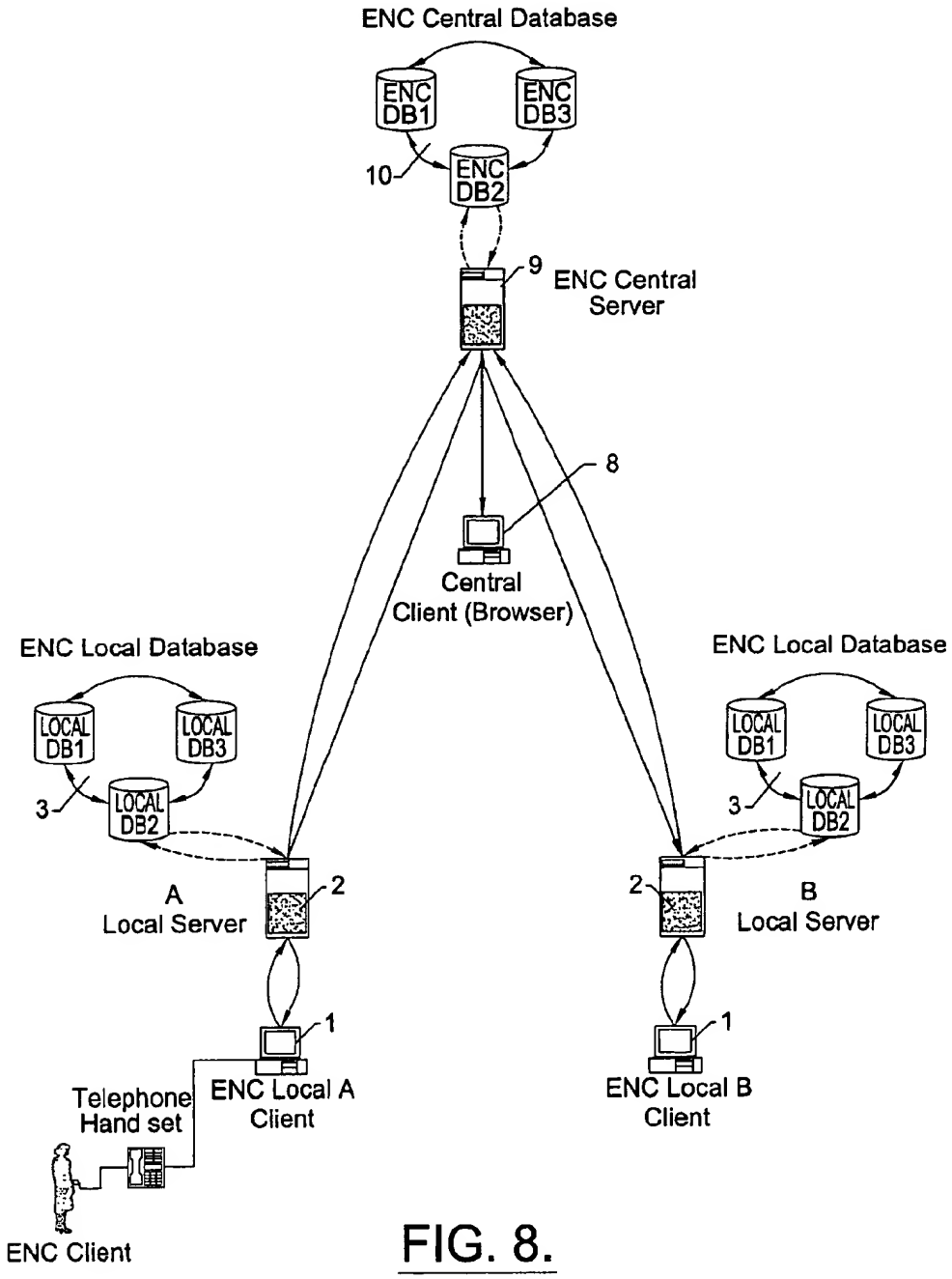

ENC Central Database
ENC DB1
ENC DB3
ENC DB2
10
9
ENC Central Server
8
Central Client (Browser)
ENC Local Database
LOCAL DB1
LOCAL DB3
LOCAL DB2
3
A Local Server
2
1
Telephone Hand set
ENC Local A Client
ENC Client
ENC Local Database
LOCAL DB1
LOCAL DB3
LOCAL DB2
3
2
B Local Server
1
ENC Local B Client

FIG. 8.

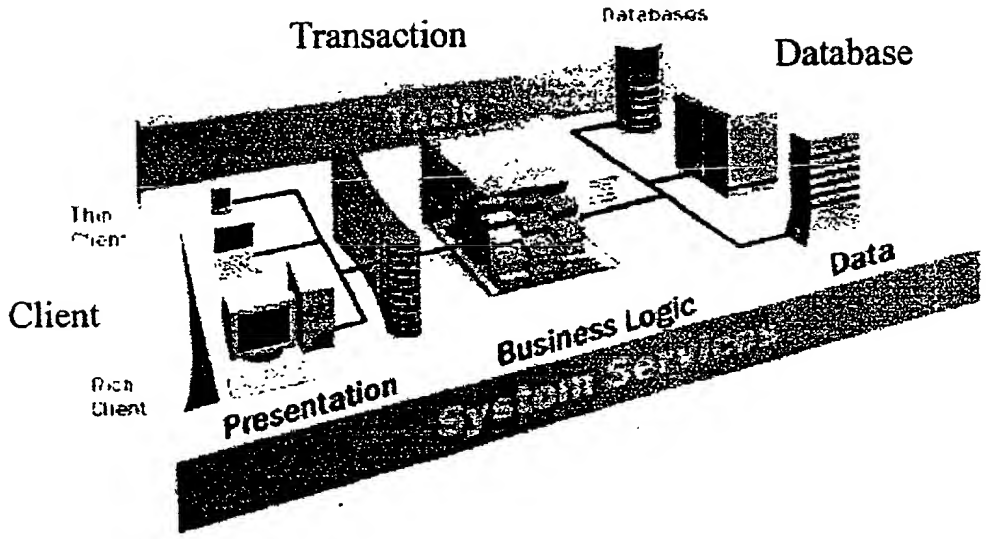

Transaction
Databases
Database
Client
Data
Business Logic
Presentation

FIG. 9.

METHOD OF IMPLEMENTING ELECTRONIC NAME CARD AND SYSTEM THEREOF

FIELD OF THE INVENTION

[0001] The present invention relates generally to the technical field of information management and exchange, especially to a method for establishing, managing and exchanging personal information using the current communications network, and a system for implementing the method.

BACKGROUND OF THE INVENTION

[0002] When doing business, most of people used to get the business identification and personal information about the traders by exchanging the business cards between each other. In case of no business card, the information about the other side is written in a notebook. This, however, will bring about a lot of inconvenience, such as

[0003] a) if there are too many business cards (names), it is inconvenient to do query and manage the cards book (address book);

[0004] b) many cards (records) would be invalid resulting from the changes, e.g., working place, of the personal status;

[0005] c) the cards book (address book) is not portable, especially when user wants to make contact with somebody in office or at home or in outdoor situation, the traditional card case or address book is extremely inconvenient;

[0006] d) in case that the address book or card case lost, the user would suffer great loss, because it is not easy to backup the business card.

[0007] Although there are some other ways of making electronic address list, such as computer, PDA etc, they still have the following limitations:

[0008] a) to establish the list, it needs user's a lot of manual inputs, which will be tedious, time-consuming and inconvenient;

[0009] b) when the information about the telephone number and address of the opposite side changes, the user has to update the address list manually;

[0010] c) various kinds of application programs, such as Word Processing Program, Fax, Mail, Browser etc, have their own address list modules that are independent, while the information needed by users is often stored in different programs so that it is hard to manage and inquire.

SUMMARY OF THE INVENTION

[0011] The objective of the invention is to provide a method for establishing virtual electronic address list, i.e. electronic name card and a system to implement the method in the current telecommunications network, so that users can establish and manage his own electronic name card without tedious input work. Users need to do nothing but dial a specific access code to make query, modification, update, exchange and telephone call.

[0012] The technical scheme achieving objectives of the invention is as follows:

[0013] A method implementing electronic name card service in telecommunications system is by means of database. It establishes virtual electronic name card in telecommunication system that includes a summation of users' personal information marked by user's ID. A User can dial a settled access number to request the electronic name card services including query, modification, exchange, and fast call.

[0014] The establishing procedure mentioned above is implemented by user's registration and applied for the service of electronic name card. It includes the steps of: a user fills in his personal information on registration; the system stores the user's information to database after authentication; at the same time, the system allocates a unique ID number to the user to identify the user in the system.

[0015] The user can build his own electronic name card case through the mentioned exchange service, which sometimes needs the authorization from the opposite side.

[0016] According to the above technical scheme, the mentioned query service includes at least the following steps:

[0017] The user dials a special service telephone number to request the service of query;

[0018] After verification of the user's identification, the system asks the user for key word of the query;

[0019] The system does the query with the key word in the local database;

[0020] If no matching record, the system will link to the central database and redo the query with the key word.

[0021] According to the above technical scheme, the mentioned modification service includes at least the following steps:

[0022] The user dials a special service telephone number to request the service of modification;

[0023] The system verifies the user's identification;

[0024] The user modifies the personal information and submits the modified form of personal information to the central database;

[0025] If the new data is accepted, the central database updates the user's record, and distributes the updated record to every local database in which the user's information is stored;

[0026] The local databases are updated in real time.

[0027] According to the above technical scheme, the mentioned exchange service includes at least the following steps:

[0028] The user dials a special service telephone number to request the service of exchange;

[0029] After verifying the user's identification, the system asks user for the query key word of the opposite side with whom the exchange will be conducted;

[0030] The system quires local database about whether there is a matching record of the electronic

name card of the opposite side. If yes, then the client will submit a transaction request of local exchange for electronic name card to the local server, and the local server will forward the request of local exchange transaction to the central server simultaneously;

[0031] The local server accepts the transaction request submitted by the client, and invokes local exchange process to deal with the transaction. At the same time, the central server accepts the transaction request submitted by the local server and invokes the exchange process of electronic name card to deal with the transaction;

[0032] If there is no matching record of electronic name card in the local database, then the system is linked to the central database and will redo the query using the query key word. If there is a matching record of electronic name card in the central database, then the local server will submit the transaction request of difference area exchange for electronic name card to the central server. The central server accepts the request from the local server and invokes the exchange process of electronic name card to deal with the transaction.

[0033] According to the above technical scheme, the fast call service mentioned includes at least the following steps:

[0034] The user dials a special service telephone number to request the service of fast call;

[0035] After verifying the user's identification, the system asks user for the query key word of the opposite side with whom the call will be conducted;

[0036] A matching record is queried in the local database with the query key word. If there is a matching record, the user's telephone will be switched to the telephone number in the matching record, so that the user can communicate directly with the desired party corresponding to the record.

[0037] An electronic name card (ENC) system implementing the above method of the invention includes at least a central transaction processing server linked with a central database system and several local transaction processing servers linked with local database systems. The central transaction processing server connects with the local transaction processing servers through the computer network. The public telecommunication network is connected with the ENC system through the local transaction processing server.

[0038] Both of the central database and the local database are all distributed system architecture.

[0039] The mentioned central database system can run on the Internet, the real time synchronization between the central database system and the local database system can be implemented through Internet.

[0040] The electronic name card provides a brand new and convenient way of communications so that the user can keep contact with all of the members recorded in the electronic name card case without manual input or remembering the telephone number of the other side. This is a special characteristic which is different from all the current ways of communications. Furthermore, although the electronic name card are same as the traditional name card, they record the user's personal information. But the information carrier of the former is paper, while the information carrier of the latter is storage equipment of electronic information, such as database etc., so that the information can be copied and exchanged more conveniently.

[0041] The benefits of the electronic name card are as follows:

[0042] 1. The traditional name card case or address book can be replaced by the electronic name card, the user can inquire his electronic name card anywhere and at anytime to communicate with the opposite side. The electronic name card will never possibly be lost and used more conveniently.

[0043] 2. It can substitute most of the address programs in the current computer. (a) the electronic name card case is created automatically saving the tedious manual input; (b) when any person information recorded in the electronic name card case changes, the user's electronic name card will update synchronously and immediately, so that failure to keep contact with the opposite side because of the changing information can be avoided; (c) with only one electronic name card case user can manage all of his communication addresses. The electronic name card case has interfaces to connect with application programs, so that the user can transmit the duplicate of the electronic name card case to computer or PDA and send fax or mail etc. conveniently.

[0044] 3. It is convenient to use ENC as easy as general telephone without any professional skills. The user needs only to memorize the access number of the electronic name card service provider (e.g. it is very easy to memorize a number like 114). By dialing the special number, it is convenient to do the followings:

[0045] (1) get the communication number of friends;

[0046] (2) inquire about the business/personal information of friends;

[0047] (3) put through the opposite side by the service provider to make a call;

[0048] (4) call the opposite side by the service provider and leave a message;

[0049] (5) modify and update his own personal/business information, and notify all the friends immediately.

THE DRAWINGS

[0050] **FIG. 1** is a flow chart for registration of ENC new user.

[0051] **FIG. 2** is a schematic flow chart of the ENC user information update.

[0052] **FIG. 3** is a schematic flow chart of the ENC user exchange.

[0053] **FIG. 4** is a schematic flow chart of the difference area exchange of the ENC user.

[0054] **FIG. 5** is a schematic flow chart of the ENC user query.

[0055] **FIG. 6** is a schematic flow chart of the ENC user fast call.

[0056] **FIG. 7** is a schematic diagram of the physical structure of the ENC system.

[0057] **FIG. 8** is a schematic diagram of the topological structure of the ENC system.

[0058] **FIG. 9** is a schematic diagram of the ENC system software with three layer structure.

DETAILED DESCRIPTION OF THE INVENTION

[0059] The invention will be described in detail with reference to the embodiments and drawings.

[0060] Electronic name card (ENC), like traditional name card, is used to record the personal information of the user. Through it, the user's identification can be found out and the user can be got in touch. ENC is a summation of the user's personal information marked by the user's ID. The implementation of ENC is by means of the database in the telecommunications system to set up the virtual electronic name card with the summation of the user's personal information marked by the user's ID. ENC provides the traditional communication users, such as telephone and handset, with a brand new communication way of service. Users can use current telephone network by dialing the special telephone number (e.g. 98998), to create, exchange, inquire ENC and communicate with the desired party.

[0061] To use ENC service, first users need to register and apply for the service at the ENC service provider. Users fill in their own personal information, and then submit to the ENC system. After validation, the ENC system will save the user information in its own database and allocate the user a unique ID number to identify the user in the system. The application can be made by filling in the registration sheet, and then making a form by manual input, the application can also be made by filling in the form on-line at ENC terminal and then submitting it to the ENC central database.

[0062] As seen in **FIG. 1**, this is a flow chart for registration of ENC new user. To apply for ENC service, first the user proposes application to the ENC service provider, inputting the name and password of ENC user. If the name of ENC user has not been used by others in the system, the system will allocate the user a unique ID number. Then the user makes his electronic name card, fills in the user's personal information including true name, telephone number, E-mail address, and the name and address of the company etc. After filling the form, it will be submitted to the ENC central database which will check the submitted form, if no mistake, then the ENC central database will create electronic name card record for the user. The system differentiates the different records by the user's ID number. At the same time, the ENC central database sends the record to the local database at the ENC service provider and requests the local database to copy the record, if this copy is successful, the registration procedure for service is finished.

[0063] After the registration procedure is finished, the personal information form is filled in and accepted by the ENC database, the user will get his own electronic name card. The electronic name card is a generic terms for the user's personal information marked by the user's ID.

[0064] ENC electronic name card can include business name card and personal name card, they can also include multimedia information and common information form, etc. what shows in Table. 1 is only an example of information definition form of database for business name card. It may vary depending upon the circumstances.

TABLE 1

Information Table of Business Name card

Field S/N	Name of field (COLUMN NAME)	Data Type	Note
1.	UserID	Int	User ID Number
2.	Name1 (FirstName)	varchar (20)	Chinese Name (English Surname)
3.	Name2 (SecName)	varchar (20)	(English Name)
4.	Company	varchar (60)	Name of Company (30 Chinese characters)
5.	CompanyFieldCode	Int	Code of Company Domain
6.	DepartmentCode	Int	Code of Company Branch
7.	Title1	varchar (16)	Title 1 (8 Chinese characters)
8.	Title2	varchar (16)	Title 2 (8 Chinese characters)
9.	Website	varchar (50)	Website URL (50 characters)
10.	BizEmail	varchar (50)	Business e-mail: a@b (a < 16, b < 30)
11.	Phone1	varchar (15)	Phone 1 (15 bit) (*extension number)
12.	Phone2	varchar (15)	Phone 2 (15 bit) (*extension number)
13.	Phone3	varchar (15)	Phone 3 (15 bit) (*extension number)
14.	Phone4	varchar (15)	Phone 4 (15 bit) (*extension number)
15.	Fax1	varchar (10)	Fax 1 (10 bit)
16.	Fax2	varchar (10)	Fax 2 (10 bit)
17.	Fax3	varchar (10)	Fax 3 (10 bit)
18.	Mphone	varchar (15)	Mobile phone (15 bit)
19.	Beeper	varchar (20)	Pager: a–b (a < 10, b < 10)
20.	BeeperType	Bit (1)	Type of Pager: Chinese (=1/ Numeric (=2)
21.	BeeperAuto	varchar (20)	Auto Calling Method (20)
22.	AreaCode	char (6)	ZIP Code (6 bit)
23.	PostCode	char (6)	Post Code of Company (6 bit)
24.	Address1	varchar (40)	Address 1 of Company (20 Chinese Characters)
25.	Address2	varchar (40)	Address 2 of Company (20 Chinese Characters)
26.	DefaultCallCode	Int	Code of Default Communication Mode
27.	CompanyCityCode	Int	Code of City Where Company is located
28.	CompanyProvinceCode	Int	Code of Province Where Company is located
29.	CompanyCountryCode	Int	Code of Nation, Default No (No Input Interface)
30.	EnableSearching	Bit (1)	Whether or not searchable (Y/N)
31.	SwitchOption	Bit (1)	Mode of Exchanging Name Card (Authorized/ At will)
32.	BizcardCreation	Bit (1)	Whether or not business name card have been made (Y/N)

[0065] The function of electronic name card is the same as that of general name card, used for marking the user's

personal information, exchanging with the other side and keeping contact between each other. The difference between them lies in: electronic name card are digital information record stored in the database of computer; traditional name card are traditional presswork in which the carrier is common paper. It is apparent that ENC has obvious predominance over the traditional name card in the aspects of query, update, replication and distribution, etc.

[0066] When the NEC user's personal information (such as number of phone or handset) changes, the user's record in the ENC database becomes out of date, the user needs to update it to ensure the information correct. As seen in **FIG. 2**, a flow chart of updating ENC user's information, the procedures are as follows:

[0067] 1) When ENC user find out that some information should be updated, he can dial the number of ENC provider and request for updating some of his own personal information. (He can directly go to ENC service provider to handle, also).

[0068] 2) After ENC service provider verifies the user's identification (password verification), the user's ID number is inputted, the user's personal information is extracted from the ENC database and displayed on the information query interface of computer (the interface is similar to that of Table 1).

[0069] 3) ENC service provider, at the user's request, modifies his personal information and refills in the personal information form.

[0070] 4) After the form modification finished, it will be submitted to the ENC central database. If no error, the renewed user's information will be written into the database to update the user's record.

[0071] 5) ENC central database (on Internet) dispenses the updated user information to every ENC local database (at the service provider) in which the user information is stored and requests these local databases to be updated in real time.

[0072] 6) If the above operations are successful, the hit message will return and the procedure is finished.

[0073] When ENC user gets the other user information about the electronic name card ID number, name and company name, etc., through his own ENC service provider, he can exchange cards with the other side. As seen in **FIG. 3**, the procedures are as follows:

[0074] 1) If the user needs to exchange electronic name card with a friend, he can dial the number of ENC service provider and request the service of exchanging name card.

[0075] 2) After ENC service provider verifies the user's identification (password verification), the user's ID number is inputted and the application program for exchanging name card is started.

[0076] 3) ENC service provider requests for query key word, such as ID number, name, etc, and does query firstly in the local database using the key word.

[0077] 4) If a matching ENC record is found in the local database, then it is determined that the other side and the user are located at the same ENC service provider. The ENC provider asks the user whether or not he hopes to exchange ENC with the other side: if the user abandons, then quit; if the user verifies the exchange, then the ENC local exchange procedure is started. If the other side requests the name card exchange must be authorized. It is only after getting the authorization from the other side, the ENC exchange service can be finished successfully.

[0078] 5) If no matching ENC record is found in the local database, then the provider sends the query request to the ENC central database with the same key word.

[0079] 6) If matching ENC record is found in the central database, then it is determined that the user and the other side does not belong to the same ENC service provider. The provider asks the user whether or not he hopes to exchange ENC with the other side: if the user abandons, then quit; if the user verifies the exchange, then the ENC exchange procedure with difference area is started. The procedure is seen in **FIG. 4**. If the other side requests name card exchange must be authorized. Then only after getting the authorization from the other side, the service of ENC exchange can be finished successfully.

[0080] 7) If no matching ENC record is found in the central database, then the provider notifies the user there is no name card for exchange.

[0081] 8) The provider asks the user whether or not he hopes to continue the service of exchange, if yes, it goes back to step 3, otherwise, the procedure of ENC exchange service is ended.

[0082] After registering as a user of ENC system, the user can create his own electronic name card case. If the other side, who the user hopes to save in his name card case, is also a registered user in ENC, by knowing the information of ID number etc. of the other side, the user can, through ENC system, add the other side into his own electronic name card case. Naturally, sometimes an authorization from the other side is needed.

[0083] The system distinguishes the electronic name card case of different users through users' ID. The creation of name card case is automatic through the above exchange procedure of the electronic name card. The exchange procedure is similar to the traditional name card exchange in the form. It is more advanced and convenient because of its basis of digital information technology.

[0084] ENC name card case can have many types, such as business name card case, friend name card case, relative name card case, and classification table of name card case. See Table 2 of business name card case, Table 3 of friend name card case, Table 4 of database definition information sheet for classification table of name card case. They may vary depending upon the circumstances.

TABLE 2

Business Name card Case

Column No	Column Name	TYPE	Note
1	UserID	Int	User ID number
2	FriendUserID	Int	Friend ID number
3	SubCategoryCode	Int	Category Code
4	Status	Int	Status(0 normal, 1 recycle bin, 2 deleted)
5	FriendName1	Varchar (20)	Name of friend (English name)
6	FriendName2	Varchar (20)	(English Surname)
7	F_Company	Varchar (60)	Name of friend's company
8	F_CompanyFieldCode	Int	Code of friend's company field
9	F_CompanyProvinceCode	Int	Code of province where friend's company is located
10	F_CompanyCityCode	Int	Code of city where friend's company is located
11	F_CompanyCountryCode	Int	Code of state where friend's company is located
12	F_DefaultCallCode	Int	Code of default call mode (with default mode of other side changed, dynamic modification is triggered)
13	Remark	Varchar (100)	Memo (50 Chinese characters)

[0085]

TABLE 3

Friend Name card Case

Column No	Column Name	TYPE	Note
1.	UserID	Int	User ID number
2.	FriendUserID	Int	Friend Id number
3.	FriendType	Int	Type of friend (0/general, 1/good friend)
4.	SubCategoryCode	Int	Category Code
5.	Status	Int	Status (0 normal, 1 recycle bin, 2 deleted)
6.	FriendName1	Varchar (20)	Name of friend (English name)
7.	FriendName2	Varchar (20)	(English surname)
8.	F_PenName	Varchar (40)	Pen name of friend
9.	F_PersonProvinceCode	Int	Code of province where friend lives
10.	F_PersonCityCode	Int	Code of city where friend lives
11.	F_PersonCountryCode	Int	Code of state where friend lives
12.	F_DefaultCallCode	Int	Code of default call mode (with the default call mode of other side changed, dynamic modification is triggered)
13.	Remark	Varchar (100)	Memo (50 Chinese characters)

[0086]

TABLE 4

Name card Case Classification

Column No	Column Name	TYPE	Note
1.	UserID	Int	User ID
2.	Category	Int	Category (1/business, 2/good friend, 3/friend, 4/member card)
3.	SubCategoryName	Varchar (20)	Name of category (10 Chinese characters)
4.	SubCategoryCode	Int	Code of category
5.	Status	Int	status (0 normal, 1 recycle bin, 2 deleted)

[0087] Wherever ENC user is, whenever he hopes to contact with a friend or know the relevant information about the friend, he can dial the number of the ENC service provider and get the needed information by using the query service of electronic name card case. As seen in **FIG. 5**, the procedures of query service are as follows:

[0088] 1) If the user needs to know the personal information about a friend, he can dial the number of the ENC service provider and request for inquiring about his own electronic name card case.

[0089] 2) After the user's identification is verified (password verification), the provider starts the query program of electronic name card case, inputs the user's ID number and enters the user interface of electronic name card case.

[0090] 3) The provider requests for query key word from the user and does the query firstly in the local database by using the key word.

[0091] 4) If a matching record is found, then the provider notifies the user the information in the matching record.

[0092] 5) If no matching record in the local database, then the provider sends the query request to the ENC central database with the same key word.

[0093] 6) If a matching record is found in the ENC central database, then the provider notifies the user the information in the matching record; if no matching record in the ENC central database, then the provider notifies the user that relevant information has not been found.

[0094] 7) The ENC service provider asks the user whether or not he hopes to continue the query procedure, if yes, then goes back to step 3, otherwise, the query procedure is ended.

[0095] Wherever ENC user is, whenever he hopes to call a friend, he may dial the number of the ENC service provider and get in touch with the opposite side to make a call by using the service of fast call. Compared with the existing communication way of telephone, ENC users needn't search in a pile of name card any more, and it is also not necessary to input long number of phone or handset. The communications procedure becomes more simple and convenient.

[0096] As seen in **FIG. 6**, the procedures of the ENC service of fast call are as follows:

[0097] 1) When the user needs to call the other side, he may dial the number of the ENC service provider and request the service of fast call.

[0098] 2) After verification of the user's ID (password verification), the ENC service provider starts the fast call program, inputs the user's ID number and enters the user interface of electronic name card case.

[0099] 3) The ENC service provider does query in the local database with the key word, such as person's name, nick name or name of company, provided by the user.

[0100] 4) If no matching record in the local database, it shows that fast call has failed, then the ENC service provider asks the user whether or not he hopes to continue the service; if yes, then goes back to step 3; if no, then the service quits, and the phone will hang up.

[0101] 5) If a matching record is found, then the ENC service provider try to transfer the call of the ENC user to the other side according to the number (handset, pager) in the matching record.

[0102] 6) If the call to the other side is put through successfully, it shows that the ENC service provider has successfully established the circuit connection between the user and the other side, then the service will be ended. ENC will quit the circuit connection, hang up his phone, but keep the communication between the user and the other side.

[0103] 7) The procedure of fast call is ended.

[0104] As seen in **FIG. 7**, the ENC system implementing the ENC functions mentioned above includes at least a central transaction processing server **9** and several local transaction processing servers **2**. The central transaction processing server **9** connects with the central database system **10** and the local transaction processing servers **2** connect with the local database systems **3**. The central transaction processing server connects with the local transaction processing server through the computer network. The public telecommunication network is connected with the ENC system through the local transaction processing server. Wherein, both the central database and the local database are distributed system architecture. The central transaction processing server may be connected with local transaction processing server though Internet. The mentioned central database system can run on the Internet, the real time synchronization between the central database system and the local database system can be implemented through Internet.

[0105] When there is a request for the service of ENC, through the client ENC system sends the user transaction request including query, information update and ENC name card exchange etc. to the service-processing server. After the transaction request is submitted from client **1** to transaction server **2**, the transaction server will invoke corresponding service processes at the server end to operate the database. Database server **3** executes the operation request from the transaction server **2**. The query result (RecordSet) will return from the database server **3** to the transaction server **2** and then to the ENC user through public telecommunication network.

[0106] The whole structure of the ENC system is showed in **FIG. 8**. It is the architecture of "distributed database +three-layer structure".

[0107] "Distributed database" refers to the structure of the ENC database system. The ENC system is composed of a central database storing all the ENC users' records and several local databases, which locate at each ENC service provider, storing only the users records of that provider. That is, each ENC local database is a subset of the ENC central database, and there are no overlapped records among the subsets.

[0108] The purpose of using distributed database is that most of the query and fast call services can be completed locally and rapidly through the operation of local database, in order to provide high performance to satisfy the user need. If all the queries are done in the ENC central database, it will result in the problems, such as system overloaded, network congestion, database lock and system crash etc.

[0109] When the record in the central (local) database changes, the local (central) database must ensure a real time, bi-directional and synchronous change to keep information consistency both in the ENC central database and local database. The ENC system ensures the information consistency by transaction processing.

[0110] "Three-layer structure" refers to the method of transaction processing. The system is divided into: presentation layer, i.e. client; business logic layer, i.e. transaction server; data layer, i.e. database server. It can be seen from the drawing that either ENC center **8**, **9**, **10** or ENC local provider **1**, **2**, **3** adopts the three-layer structure as basic architecture, that is client ↔ transaction server ↔ database server.

[0111] It is noted that the concept of client and transaction server is not absolute. For example, when ENC local provider needs to query about information in the ENC central database (if the user needs some uncommon information that is not stored in the ENC local provider), the ENC local transaction server is a client at this time. It sends the transaction request to the ENC central transaction server, and asks the ENC central transaction server responses the request and returns result (RecordSet) to the ENC local transaction server. When updates take place in the ENC central database and it is necessary for the ENC local database to update synchronously, the ENC central transaction server is a client at this time, it sends the transaction request to the ENC local transaction server, and asks the ENC local transaction server responses the request and returns the updating result message, success or failure, to the ENC central transaction server.

[0112] The software of the ENC system is showed in **FIG. 9**. It is three-layer software architecture oriented to the transaction processing.

[0113] The first layer is presentation layer, which is a user-machine interface layer, used by the client to receive the user event request, send the transaction request to the server and display the result returned from the server to the user. Taking a real system based on Microsoft DNA tech-

nology as example, the client can be MS IE5.0 browser running on Windows 98, the presentation layer here is the HTML web page displayed in the IE browser to the user. Browser receives the user's request by HTML form, submits the request to the web server and displays the HTML pages returned from the server.

[0114] The second layer is business logic layer. It is a transaction-processing layer, that is used for receiving the transactions submitted from the client, calculating according to the predefined conditions of the business logic and transaction, doing query in the database. It ensures the transaction complete and successful, and returns the operation result to the client. For instance, when transaction server, running on the service end, receives the transaction request submitted by browser through web server, it will invoke the corresponding transaction processing process to do the calculation and query in the database. Then, it will transfer the result of query and calculation, by transforming it into HTML pages, to the browser. Taking the above real system based on Microsoft DNA technology as example, where Microsoft Transaction Server (MTS), used as transaction-processing server, receives the transaction request submitted by IE browser through Internet Information Server (IIS), WWW server. MTS will invoke the corresponding service process component (COM) to do the service, to query in the database, and will return the result, by transforming it into HTML page to IE browser.

[0115] The third layer is data layer, i.e. database. It is used to receive the query request transmitted from the transaction processing server, search in the database according to the query conditions, and return the matching record (Record-Set) to the transaction server. Taking also the above real system based on Microsoft DNA technology as example, where SQL server database receives the query request that comes from MTS by invoking the COM component. Then it will do the query in the database by using structure query language (SQL) or invoking stored procedure and return the record set to the MTS.

[0116] The method of the above transaction processing will be further described by taking the software function of the exchange service of ENC name card as an example.

[0117] Exchanging ENC name card, i.e. user A requests for exchanging name card with user B, will have finished by a series of database operations. For example, in the software system of database, the procedure of exchanging name card between user A and B can be implemented by the following operations:

[0118] 1. Taking the user B's information field from the database;

[0119] 2. Adding a new record in the user A's electronic name card case, TABLE;

[0120] 3. Storing the user B's information field into the corresponding field of user A's new record;

[0121] (Through the above steps, user B has been added in the user A's electronic name card case.)

[0122] 4. Through the same steps as above 1~3, user A can also be added in the user B's electronic name card case.

[0123] The action of the above serial operations, used to finish a concrete event, is called "Transaction". The characteristic of transaction processing is "all is done" or "all is undone", that is to say, in the exchange service of ENC name card, only when the above four steps of database operations are totally successful, then the exchange transaction is successfully, and all the databases are updated. If there is a failure in any one of the four steps, the failure message returns immediately, and the exchange transaction of ENC name card is failed. All databases will scroll back to the initial states before the transaction starts, and i.e. all the databases are not updated.

[0124] The procedure of software processing for the different area exchange transaction of ENC name card is as follows:

[0125] ENC name card exchange transaction is divided into three small transactions to implement, i.e.

[0126] 1. ENC central server requests for exchanging ENC with ENC local server B; User A (central)↔User B (local)

[0127] 2. ENC exchange between user A and B takes place inside the ENC central server; User A (central)↔User B (central)

[0128] 3. ENC central server requests for exchanging ENC with ENC local server A; User A (local)↔User B (central)

[0129] The implementation of above 1-3 steps is a series of access operations in database. Taking step 1 as an example, user A in central database exchanges ENC with user B in local database:

[0130] 1. Reading out user B's record (column value) from the local database;

[0131] 2. Adding a new record in table of user A's ENC name card case in the central database;

[0132] 3. Writing the user B's information fields, which come from the local database B, into the corresponding fields of user A's new record in the central database;

[0133] 4. Reading out user A's record (column value) from the central database;

[0134] 5. Adding a new record in table of user B's ENC name card case in the local database;

[0135] 6. Writing the user A's information fields, which come from the central database, into the corresponding fields of user B's new record in the local database.

[0136] If the above six steps of database operations are finished successfully, the function will return a successful message, then the exchange transaction processing between user A (central) and user B (local) will be completed successfully. Otherwise, if there is failure in any one of the above steps, the operation function will return failure message, it means that the transaction processing failed; database will scroll back to the initial states before the transaction processing starts. Apparently, only when all three small transaction are finished in order, the exchange transaction of ENC name card will be successful. Otherwise failed, all has not been done. This is the method of transaction processing

for implementing ENC name card exchange. In the whole ENC software system design, this design idea is carried out all the time.

[0137] When the invention is implemented in the telecommunications network, it is more convenient to use intelligent network mode with a new intelligent business service. Other modes, such as intelligent platform or calling center may also be adopted. They are only different implementation mode and should be fallen into the protection scope defined by the invention.

[0138] The above descriptions, which are the preferred embodiments of the invention, should not limit the invention. Any modification, equivalent substitution, improvements etc, within the spirit and principle of the invention, should be included in the claims of the invention.

1. A method for implementing the service function of electronic name card in the telecommunication system, wherein it is characterized that the method includes at least:

Establishing virtual electronic name card with a summation of user's personal information marked by user's ID in the form of database in the telecommunications system;

By dialing a special telephone number so that user can request electronic name card services including query, modification, exchange, and fast call:

2. The method of claim 1 wherein it is characterized that: the said establishment is implemented by user registration and application for the services of electronic name card;

It is included that user fills in the personal information when registration; after verification, the system writes down the user's information into the database and allocates the user a unique ID number to identify the user in the system.

3. The method of claim 1 wherein it is characterized that: the user can create his own electronic name card case through the said exchange service.

4. The method of claim 1 or 3 wherein it is characterized that: the said exchange service must be further authorized by the other side to be exchanged with.

5. The method of claim 1 wherein it is characterized that the said query service includes at least the following steps:

The user dials a special telephone number to request a query service;

After verifying the user's identification, the system asks the user for query key word;

The system does the query with the key word in the local database;

If no matching record, the system will redo the query with the key word by linking to the central database.

6. The method of claim 1 wherein it is characterized that the said modification service includes at least the following steps:

The user dials a special telephone number to request a modification service;

The system verifies the user's identification;

The user modifies the personal information and submits the modified form of personal information to the central database;

The central database updates the user's record if the new data is accepted, and dispenses the updated record in every local database in which the user's information is stored;

The local database is updated in real time.

7. The method of claim 1 wherein it is characterized that the said exchange service includes at least the following steps:

The user dials a special telephone number to request a service of exchange;

After verifying of the user's identification, the system asks the user for the query key word of the opposite side to be exchanged with;

The system queries local database about whether there is a matching record of the opposite side or not;

If there is a matching record of electronic name card in the local database, then the client will submit a transaction request for local exchange to the local server, at the same time the local server will also submit a transaction request for local exchange to the central server;

The local server accepts the transaction request submitted by client, and invokes local exchange process of electronic name card to deal with the transaction, at the same time, the central server accepts the transaction request submitted by local server and invokes exchange process of electronic name card to deal with the transaction;

If there is no matching record of electronic name card in the local database, then the system is linked to the central database and will redo the query using the query key word, if there is a matching record of electronic name card in the database, then the local server submits a transaction request for difference area exchange of electronic name card to the central server, the central server accepts the transaction request from the local server and invokes exchange process of electronic name card to deal with the transaction.

8. The method of claim 1 wherein it is characterized that the said fast call service includes at least the following steps:

The user dials a special telephone number to request a fast call service;

After verifying the user's identification, the system asks the user for the query key word of the opposite side to be called;

With the query key word, a matching record is queried in the local database, If there is, the user's telephone will be transferred to link with the telephone number in the matching record, so that the user can communicate directly with the desired party corresponding to the record.

9. An electronic name card (ENC) system wherein it is characterized that:

The system includes at least a central transaction processing server and several local transaction processing servers, the central server links with central database system, and the local server with local database system;

The central transaction processing server connects with the local transaction processing server through the computer network, while the public telecommunication network is connected with the ENC system through the local transaction processing server.

10. The system of claim 9 wherein it is characterized that: both the central database and the local database uses distributed system architecture.

11. The system of claim 9 wherein it is characterized that: the said central database system runs on the Internet, the real time synchronization between the central database system and the local database system is implemented through INTERNET.

* * * * *

US 20040066397A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2004/0066397 A1

Walker et al. (43) Pub. Date: Apr. 8, 2004

(54) COMPUTER NETWORK HAVING CONTEXT SENSITIVE APPLICATIONS AND CONTROLS FORMING DYNAMIC USER INTERFACES ON LOCAL COMPUTER TERMINALS

(76) Inventors: **Bradley K. Walker**, Lincoln, NE (US); **John D. Turnipseed**, Lincoln, NE (US); **Daniel J. Castagnoli**, Lincoln, NE (US)

Correspondence Address:
BLACKWELL SANDERS PEPER MARTIN LLP
TWO PERSHING SQUARE
2300 MAIN STREET, SUITE 1000
KANSAS CITY, MO 64108 (US)

(21) Appl. No.: **10/679,565**

(22) Filed: **Oct. 6, 2003**

Related U.S. Application Data

(62) Division of application No. 09/789,904, filed on Feb. 21, 2001.

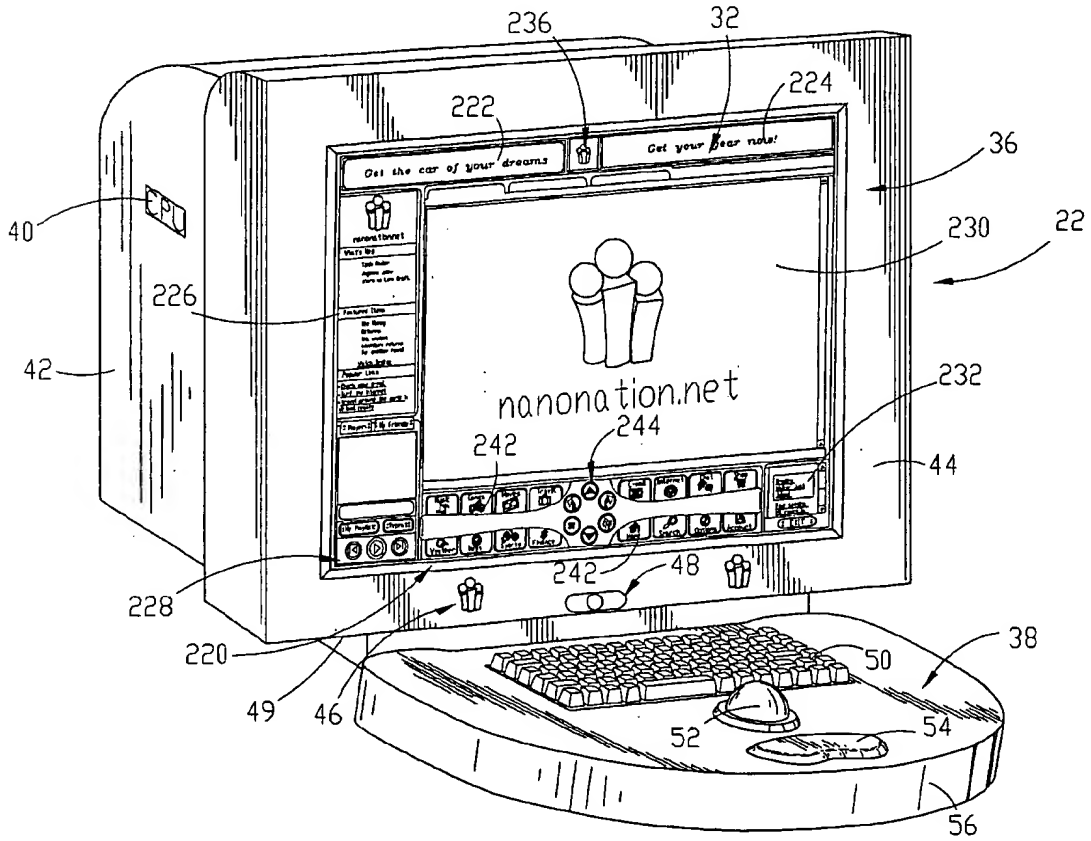
Publication Classification

(51) **Int. Cl.**[7] **G09G 5/00**

(52) **U.S. Cl.** **345/719**

(57) **ABSTRACT**

A computer network (**20**) having a plurality of terminals (**22**) and several network servers (**26, 28, 30**) are operative to develop context-sensitive, dynamic graphical user interfaces (**32**) which are programmed centrally by the network servers (**26, 28, 30**). XML packets (**142, 144**) are used to transfer information regarding the graphical user interfaces (**32**) between the terminals (**22**) and the network servers (**26, 28, 30**). The graphical user interface (**32**) is a layered multimedia environment having a background movie (**248**) played beneath substantially all of a plurality of screen application regions (**220-230**). A interactive control movie (**240**) is displayed in one of the screen application regions and includes control tabs (**242**) and dynamic button controls (**244**). A browser application (**250**) may be embedded in the background movie (**248**) as part of the layered media environment. Multiple applications can be operated in the selected screen application region, so each application is provided with a top tab (**246**) enabling users to bring the desired application to the top of the application panels.



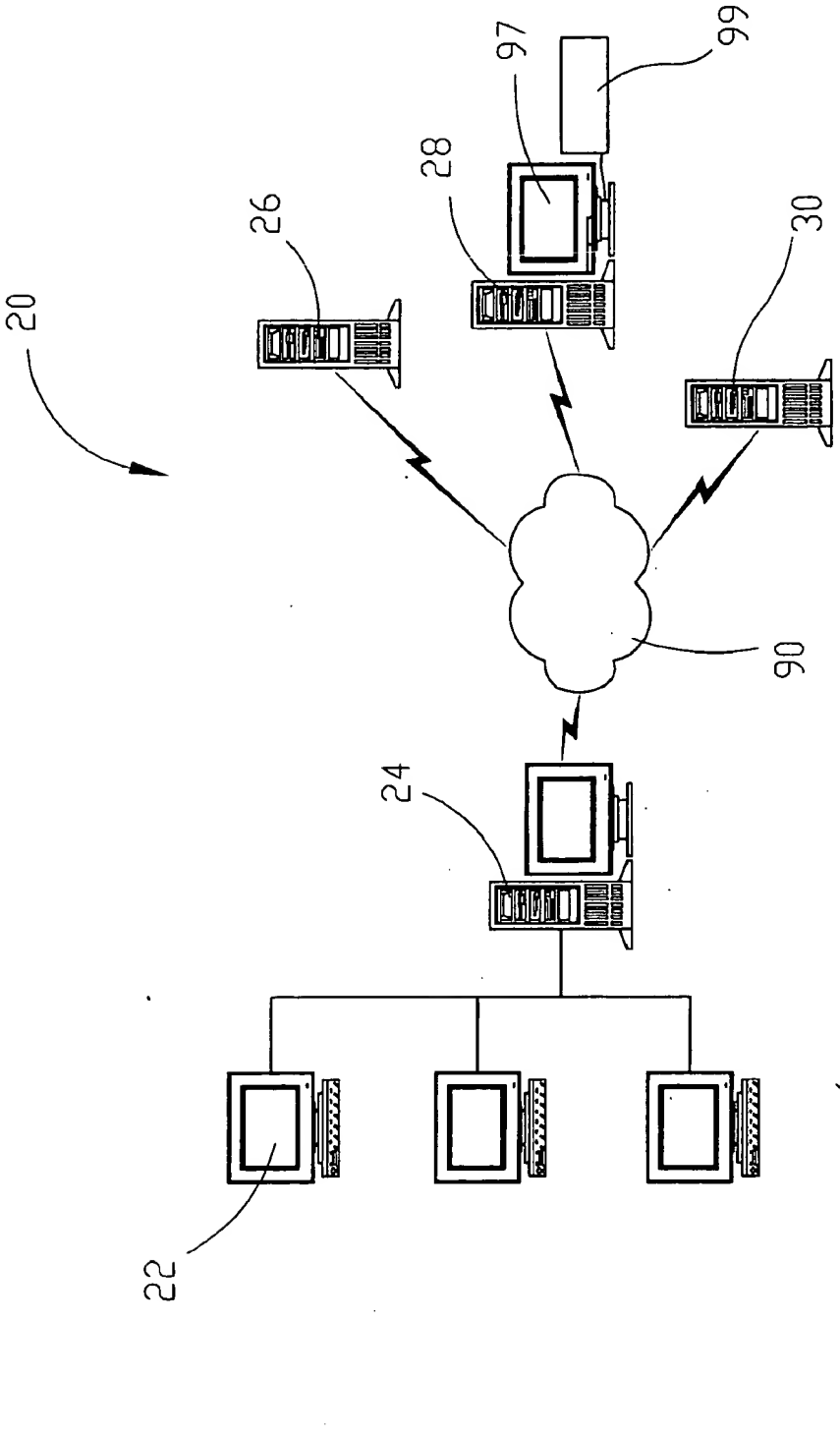


Fig. 1.

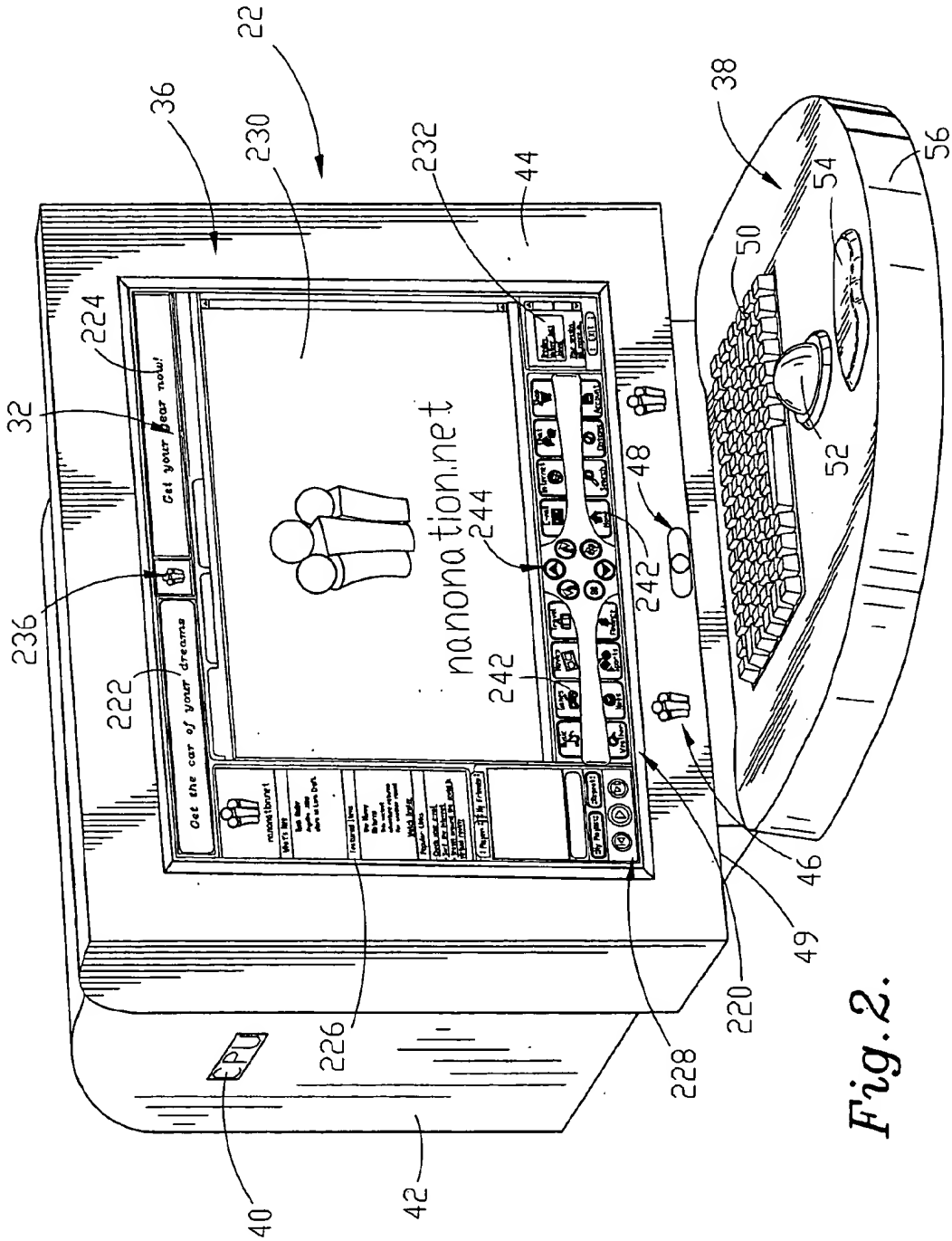

22
36
230
232
44
38
56
54
50
52
224
32
48
244
242
236
222
242
46
49
220
226
228
40
42

Fig. 2.

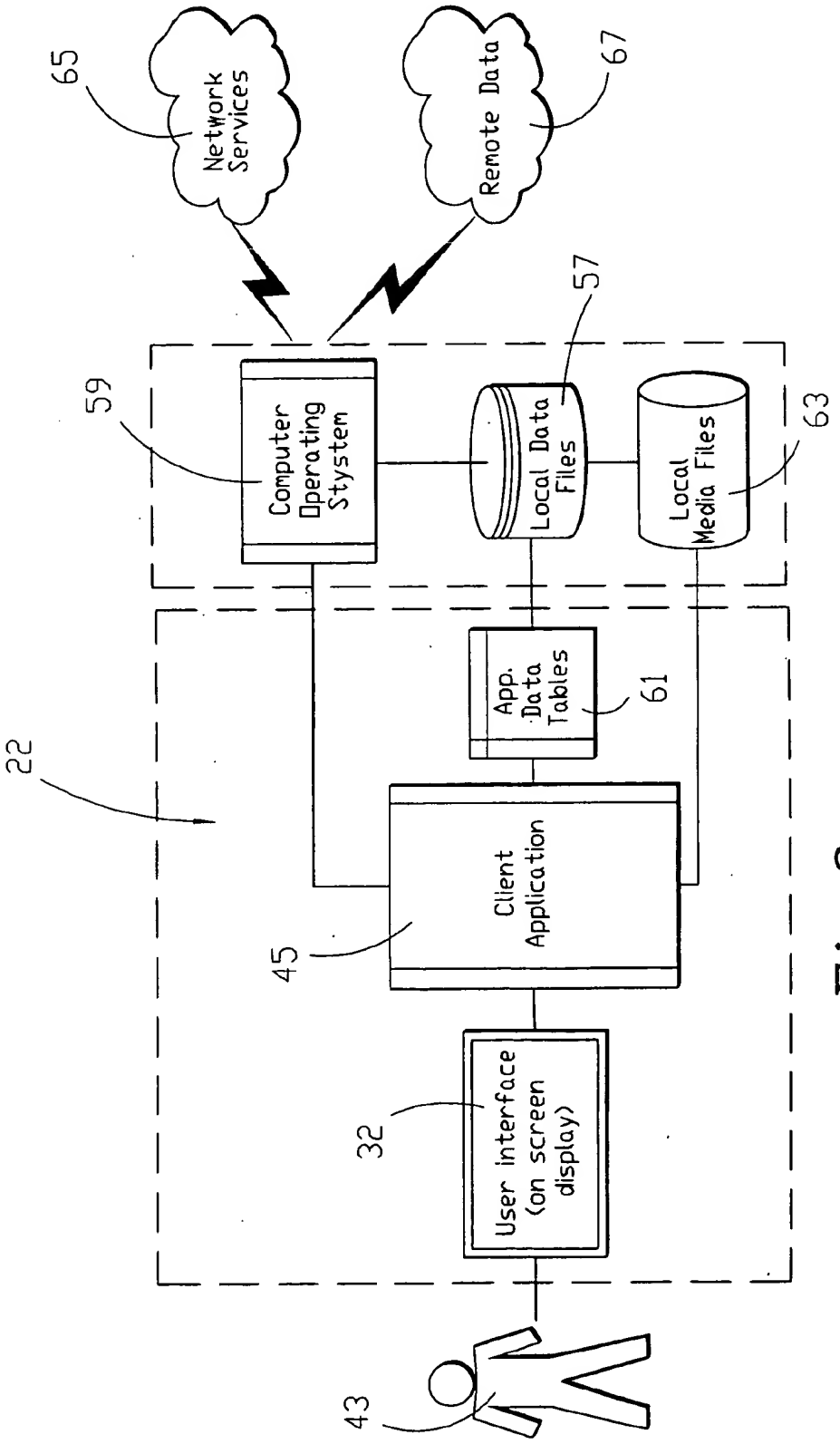

65
Network Services
67
Remote Data
57
59
Computer Operating Stystem
Local Data Files
Local Media Files
63
App. Data Tables
61
22
Client Application
45
User interface (on screen display)
32
43

Fig. 3.

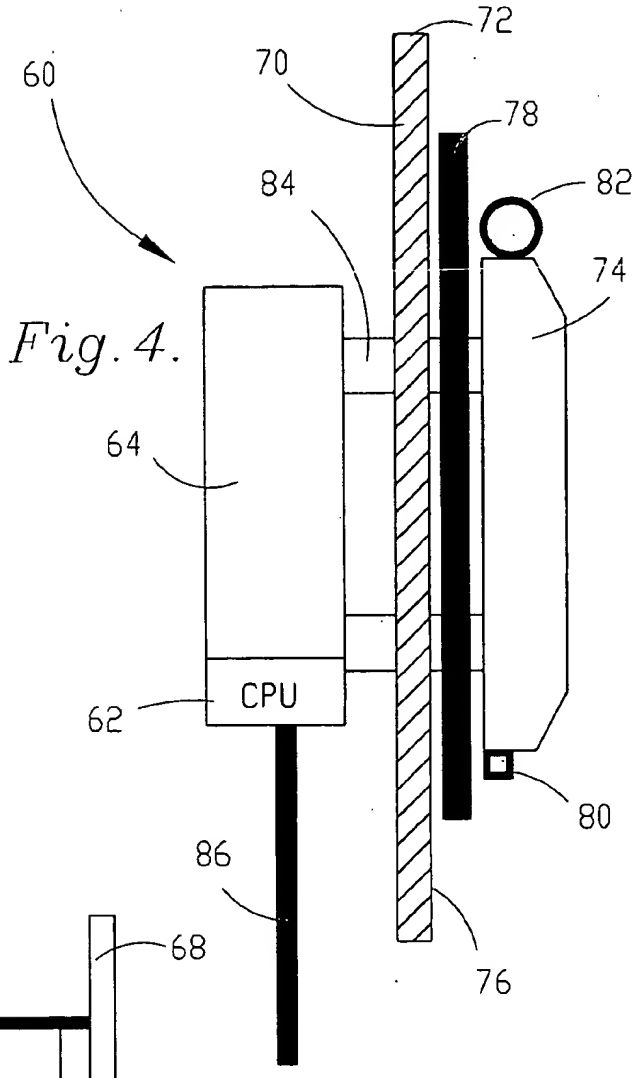


Fig. 4.

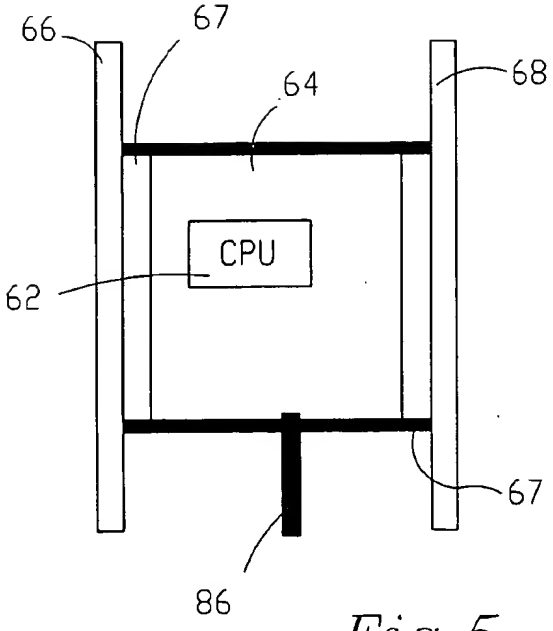


Fig. 5.

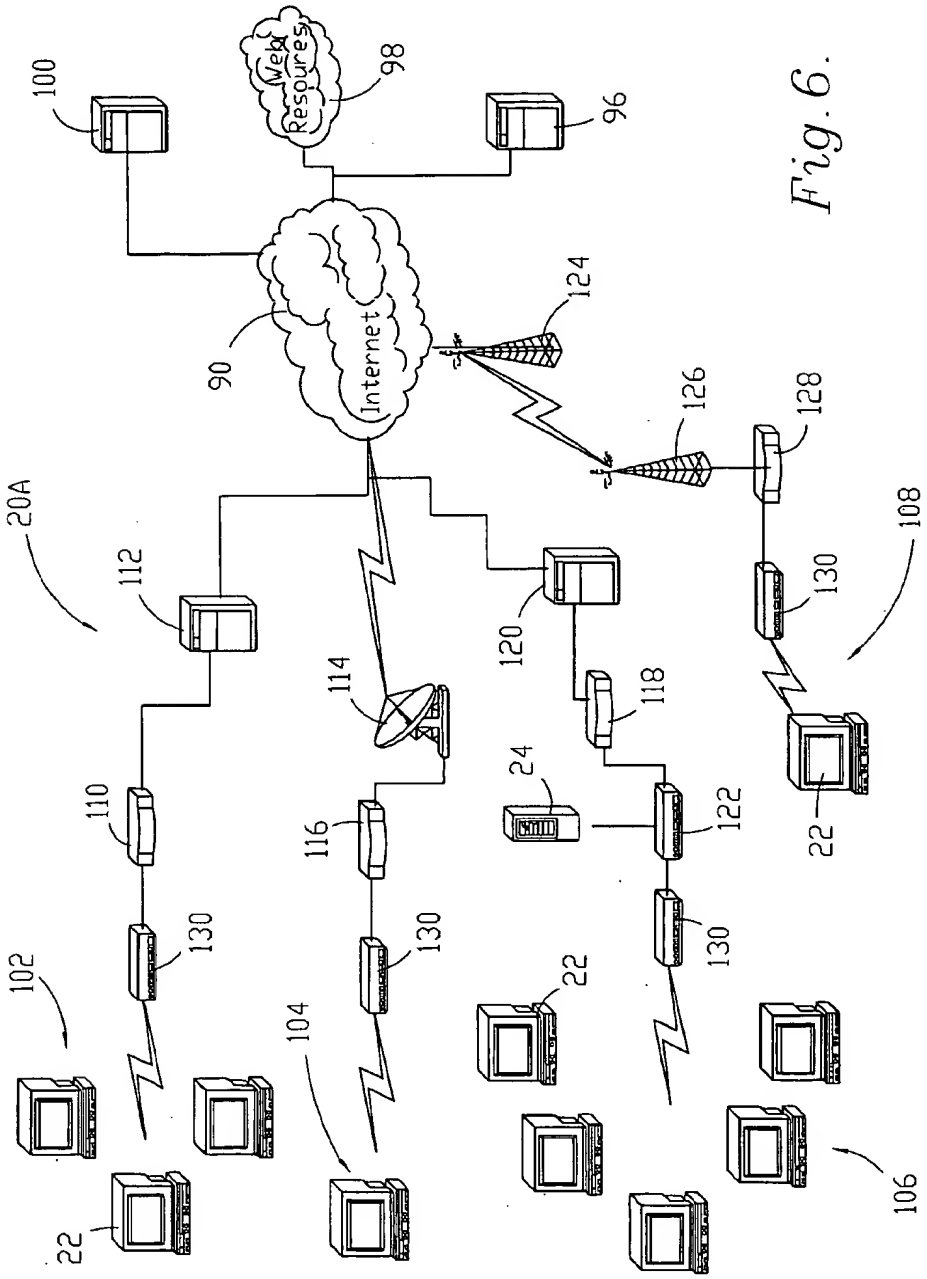

Web Resources
98
100
96
90
Internet
124
126
128
20A
112
120
108
130
114
118
24
110
116
122
22
130
130
130
102
104
22
22
106

Fig. 6.

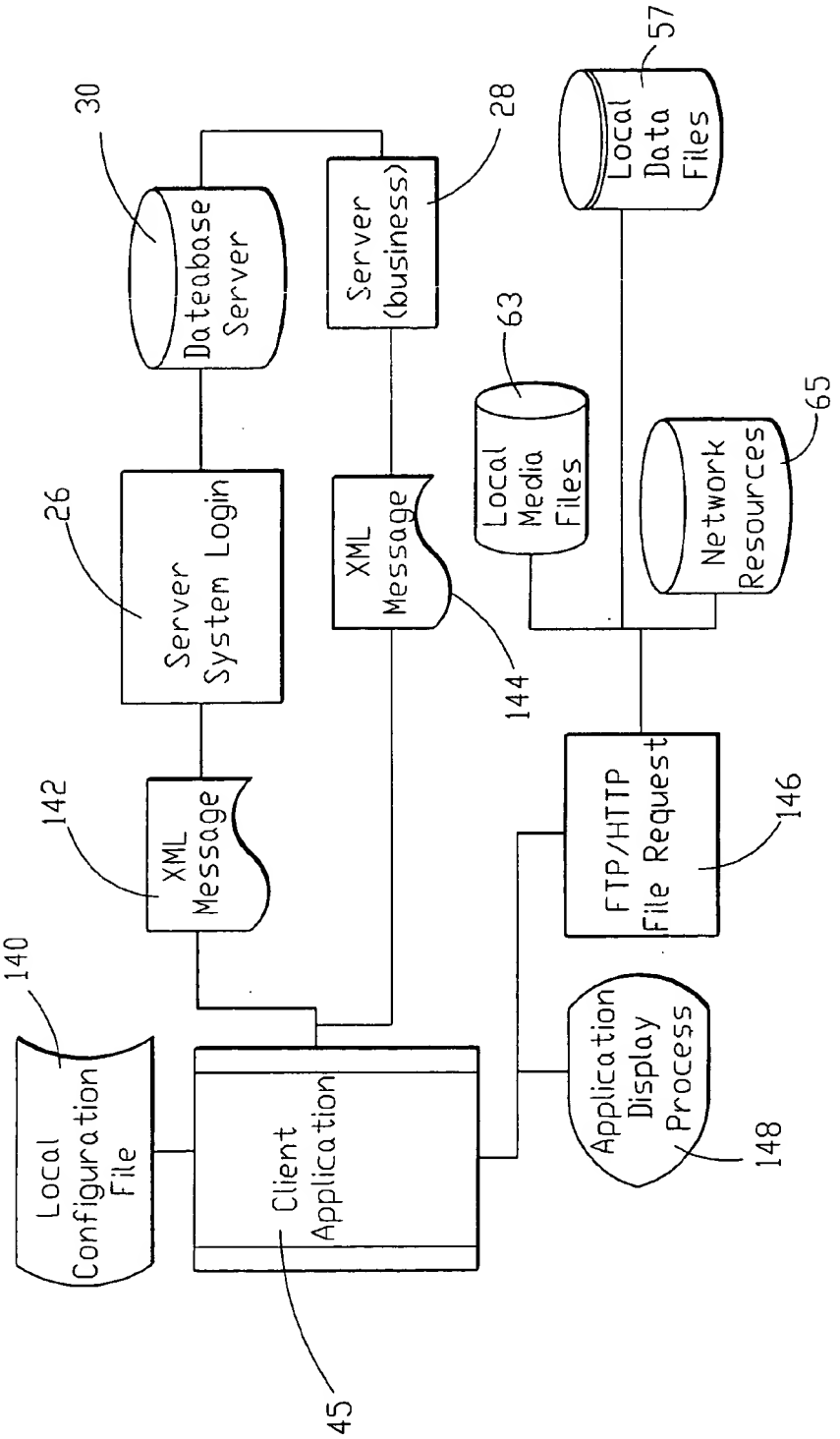

30
Database Server
28
Server (business)
57
Local Data Files
63
Local Media Files
65
Network Resources
26
Server System Login
XML Message
144
142
XML Message
FTP/HTTP File Request
146
140
Local Configuration File
Client Application
Application Display Process
148
45

Fig. 7.

Fig. 8.

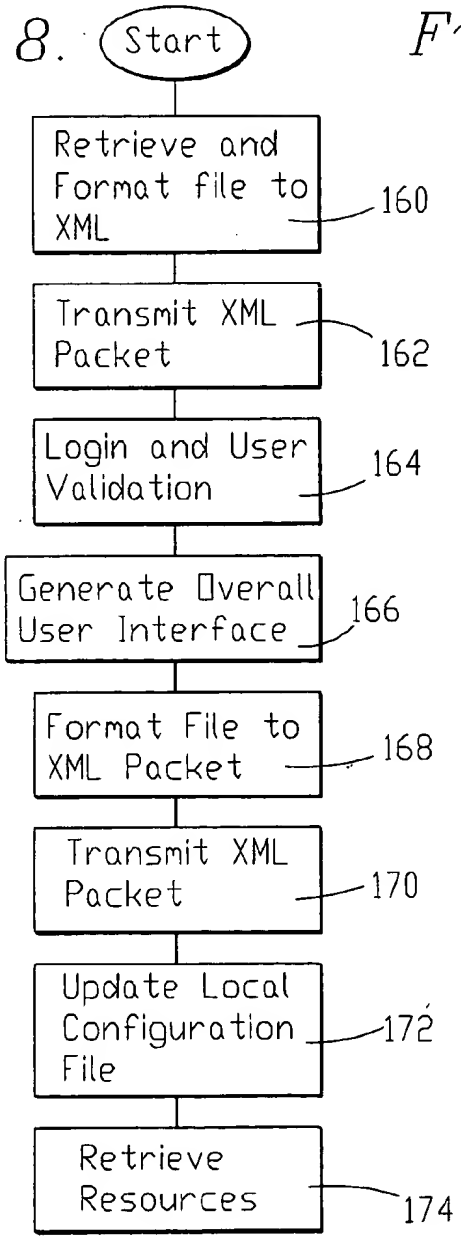

Start
Retrieve and Format file to XML
160
Transmit XML Packet
162
Login and User Validation
164
Generate Overall User Interface
166
Format File to XML Packet
168
Transmit XML Packet
170
Update Local Configuration File
172
Retrieve Resources
174

Fig. 9.

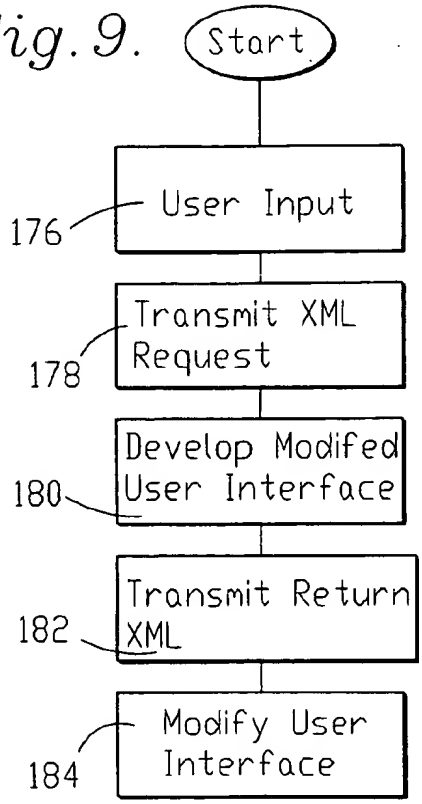
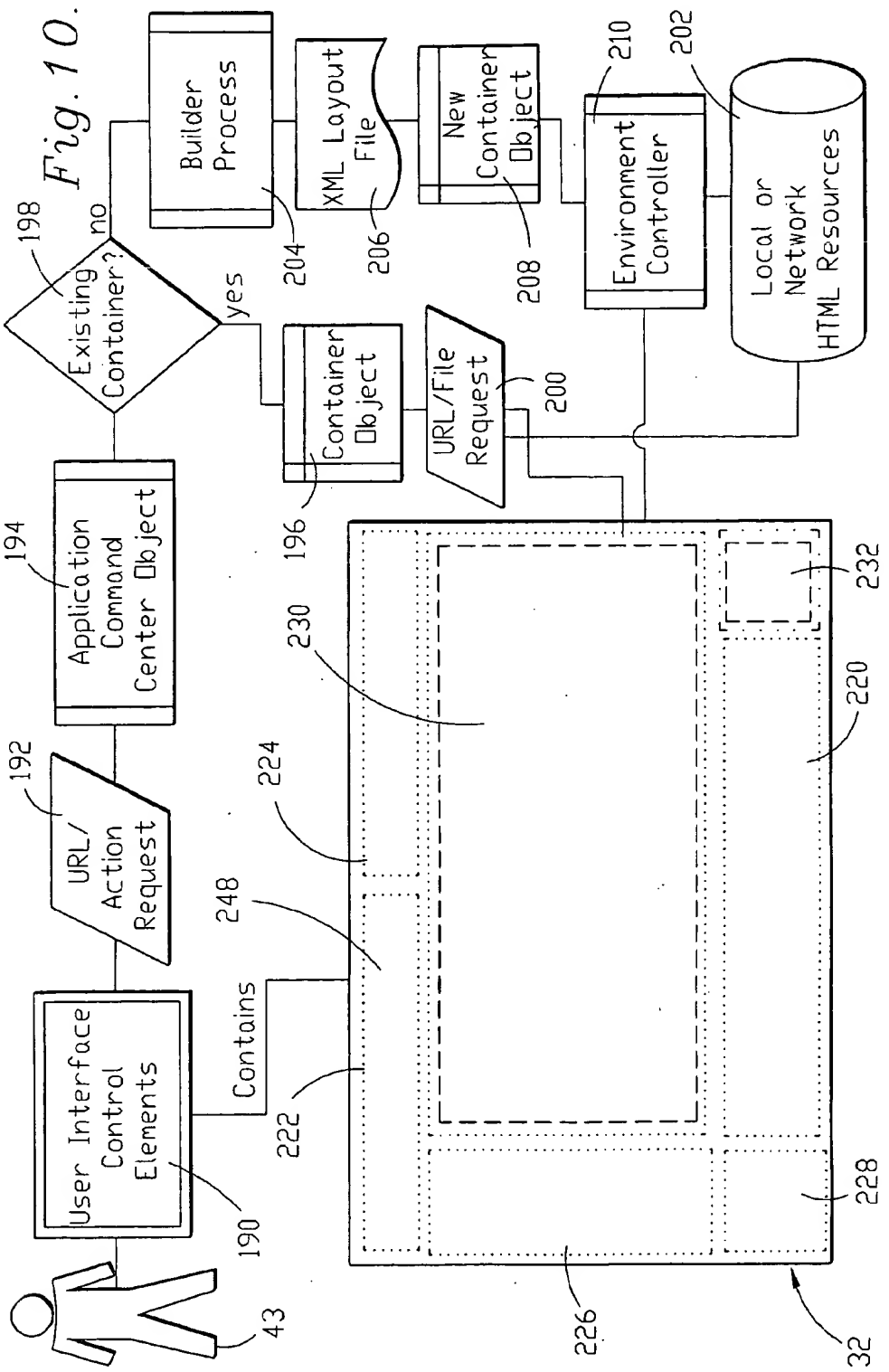

Start
176
User Input
Transmit XML Request
178
Develop Modifed User Interface
180
Transmit Return XML
182
Modify User Interface
184

Fig. 10.


Builder Process
XML Layout File
New Container Object
Environment Controller
Local or Network HTML Resources
Existing Container?
no
yes
Container Object
URL/File Request
Application Command Center Object
URL/ Action Request
User Interface Control Elements
Contains
198
204
206
208
210
202
200
196
194
192
190
230
232
220
224
248
222
228
226
43
32

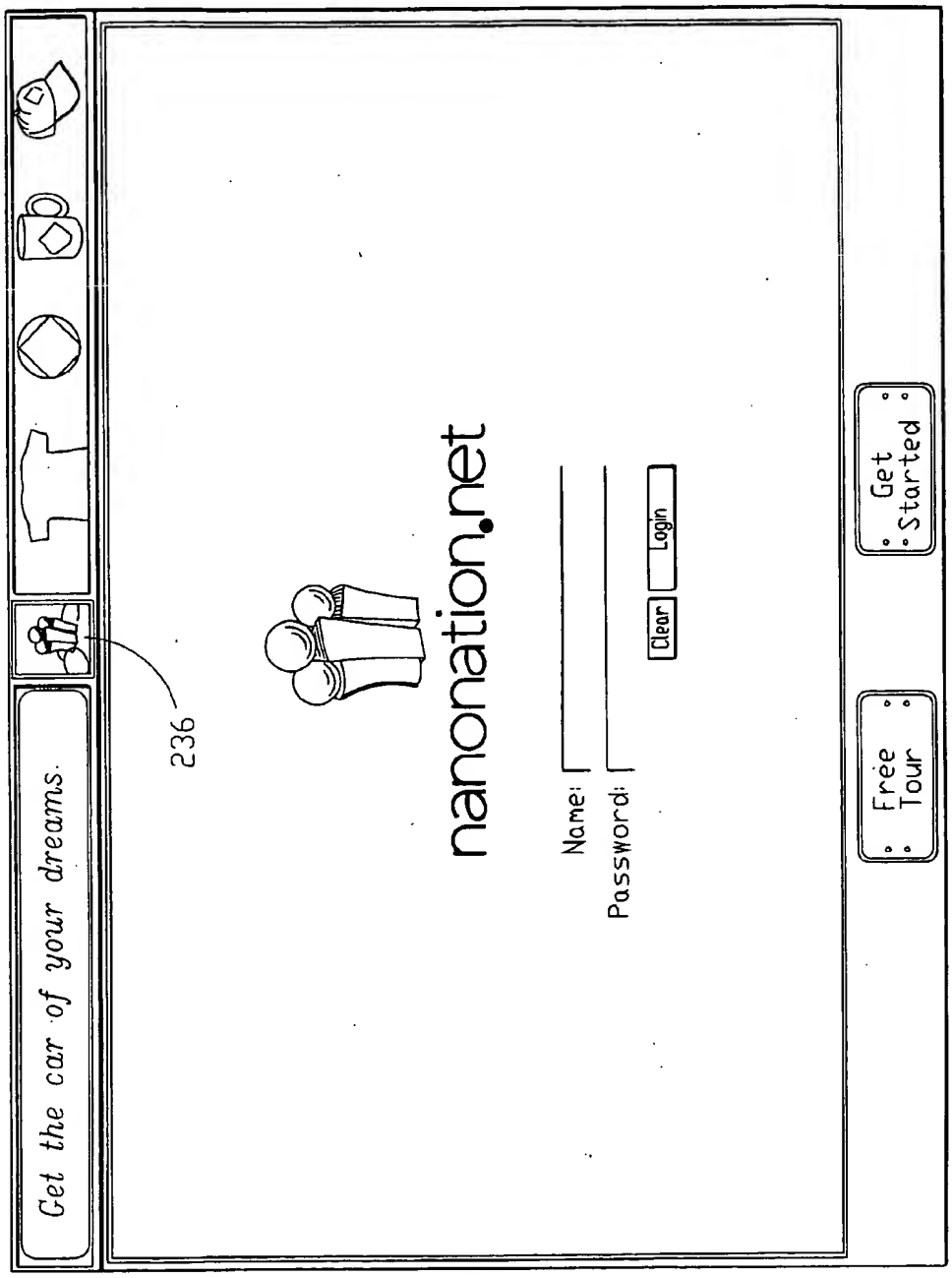

Get the car of your dreams.
236
nanonation.net
Name:
Password:
Clear
Login
Free Tour
Get Started

Fig. 11A.

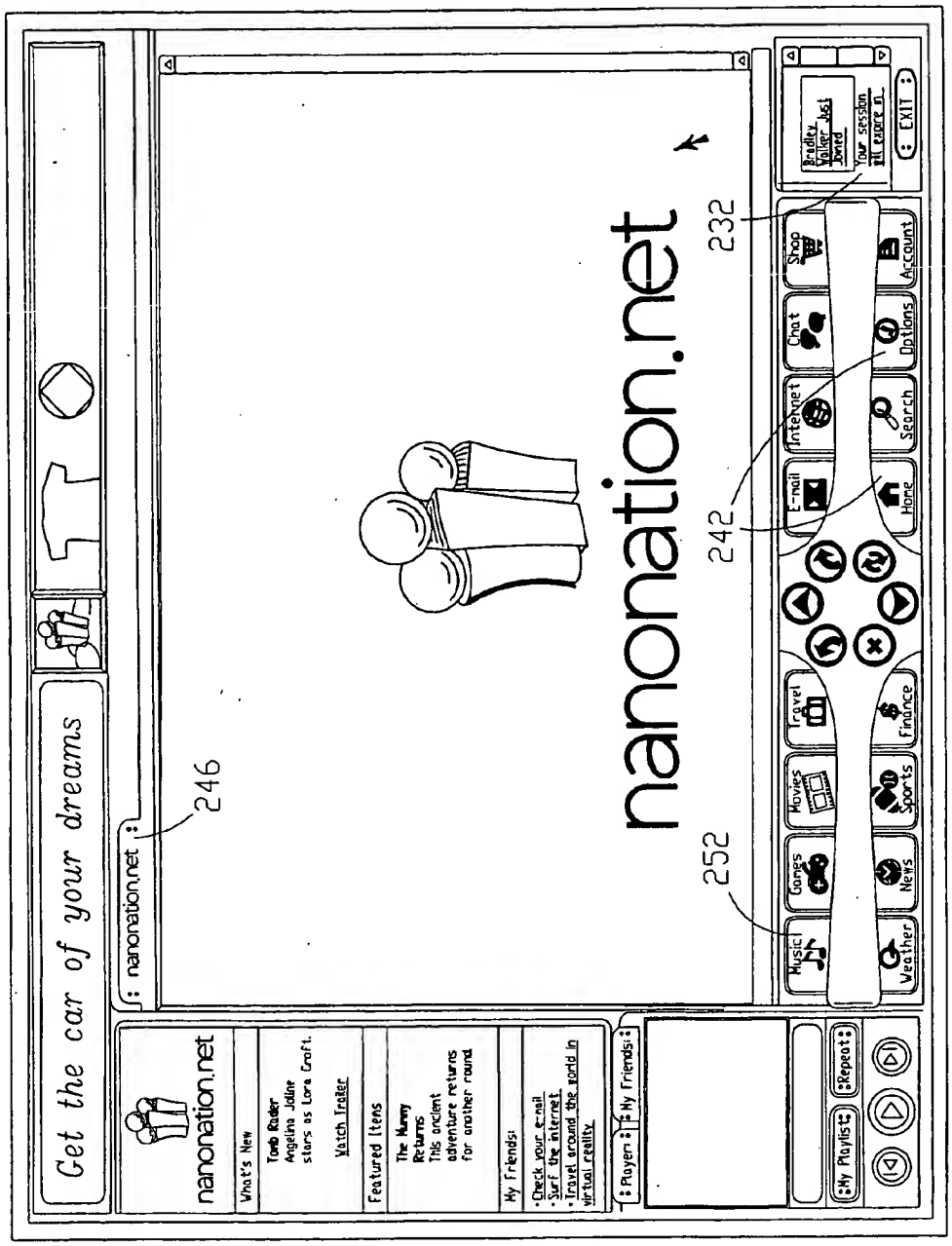

Get the car of your dreams
nanonation.net
What's New
Tomb Raider
Angelina Jolie
stars as Lara Croft.
Watch Trailer
Featured Items
The Mummy
Returns
This ancient
adventure returns
for another round
My Friends:
•Check your e-mail
•Surf the internet
•Travel around the world in
virtual reality
: Player :
: My Friends :
: nanonation.net :
246
nanonation.net
232
242
252
Music!
Games
Movies
Travel
Weather
News
Sports
Finance
E-mail
Internet
Search
Home
Chat
Options
Shop
Account
: EXIT :
Bradley
Walker Joel
Bored
Your session
ID expire in
:My Playlist:
:Repeat:

Fig. 11B.

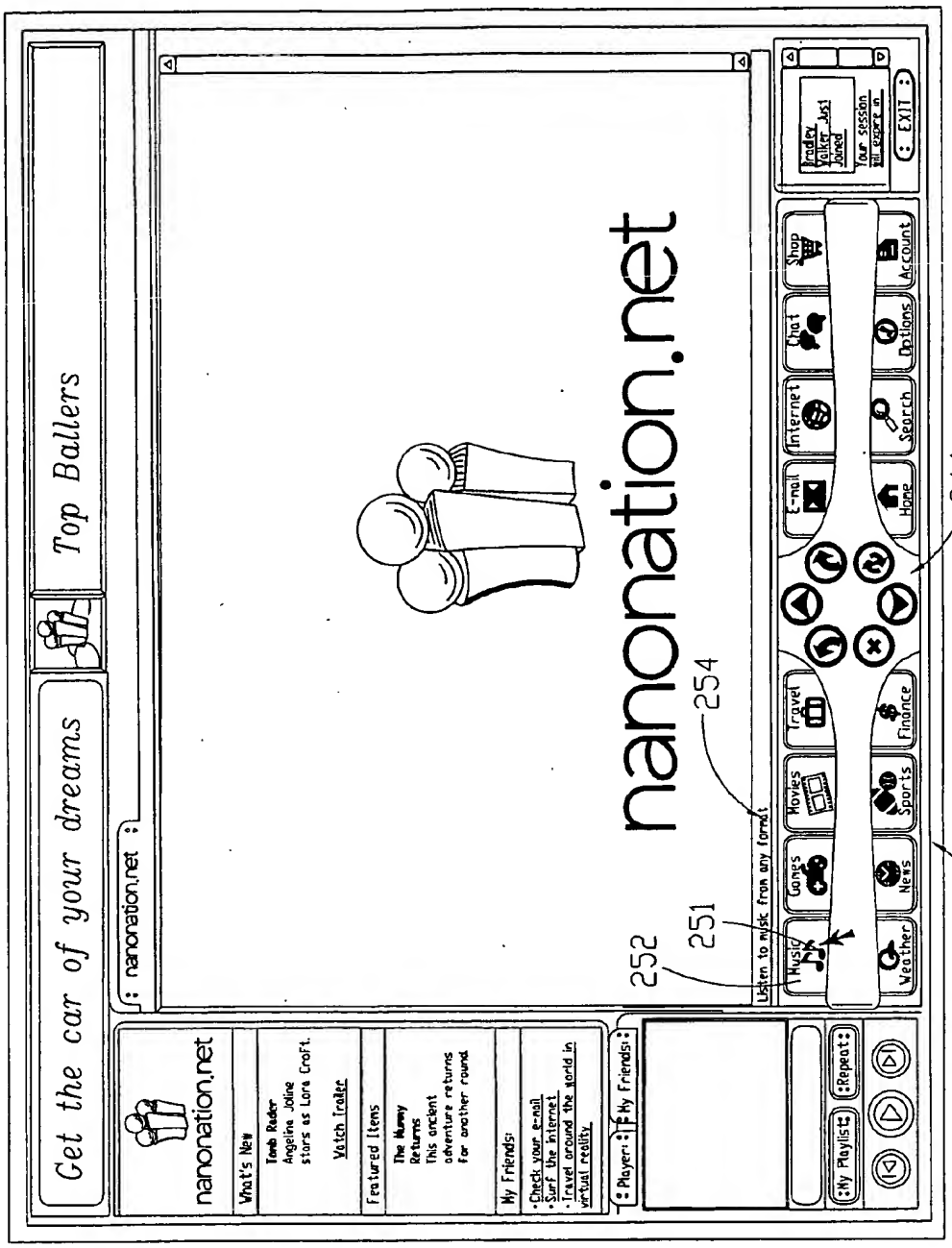

Get the car of your dreams
Top Ballers
nanonation.net
What's New
Tomb Raider
Angelina Jolie
stars as Lara Croft.
Watch Trailer
Featured Items
The Mummy
Returns
This ancient
adventure returns
for another round
My Friends:
•Check your e-mail
•Surf the internet
•Travel around the world in
virtual reality
Player:
My Friends:
My Playlist:
Repeat:
nanonation.net
Listen to music from any format
Music
Games
Movies
Travel
Sports
Finance
News
Weather
E-mail
Internet
Chat
Shop
Home
Search
Options
Account
EXIT
252
251
254
240
244

Fig. 11C.

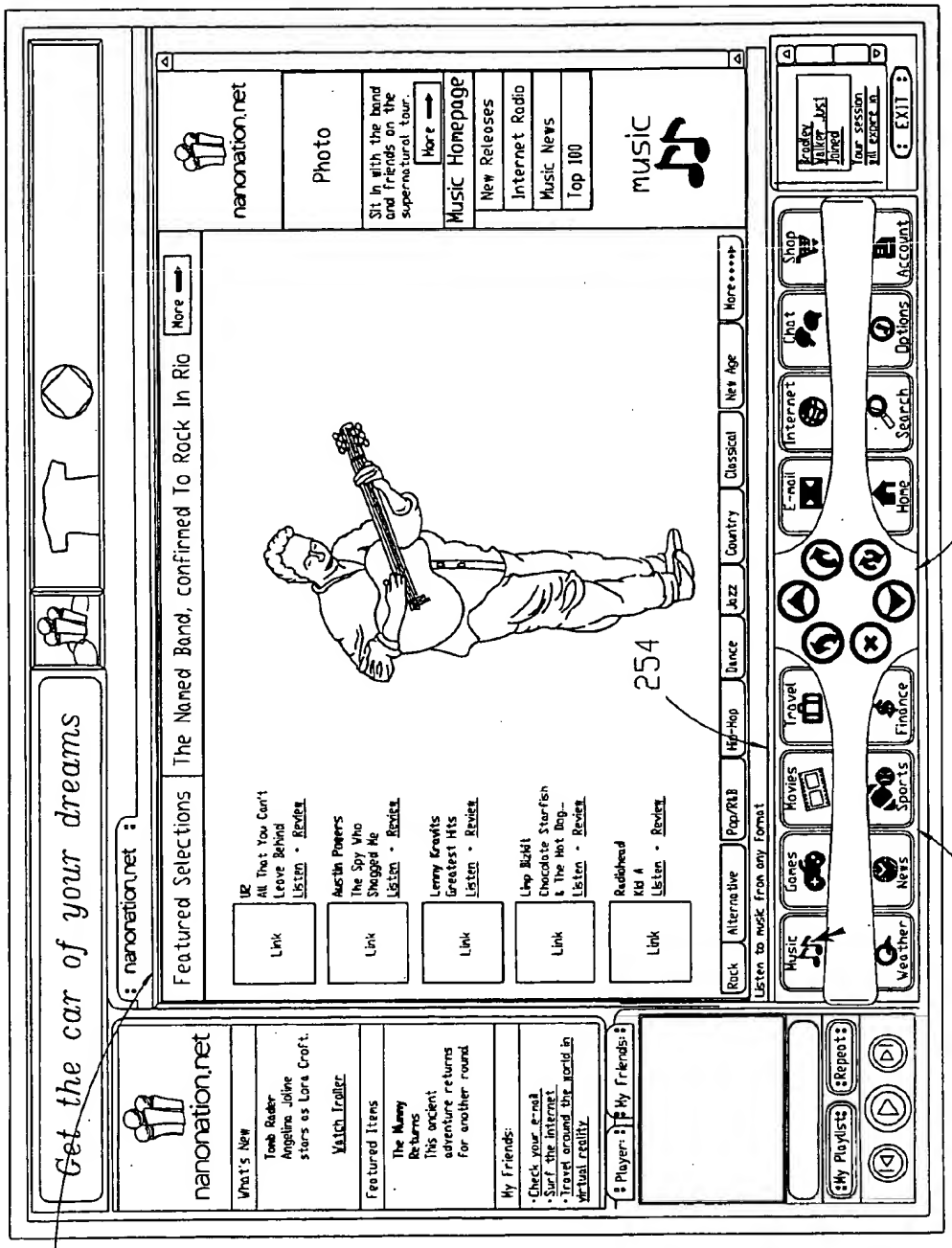

Get the car of your dreams
nanonation.net
Featured Selections
The Named Band, confirmed To Rock In Rio
More
Photo
Music Homepage
New Releases
Internet Radio
Music News
Top 100
music
Rock
Alternative
Pop/R&B
Hip-Hop
Dance
Jazz
Country
Classical
New Age
Music
Games
Movies
Travel
Sports
Finance
Weather
News
E-mail
Internet
Chat
Shop
Home
Search
Options
Account
EXIT
258
254
240
244

Fig. 11D.

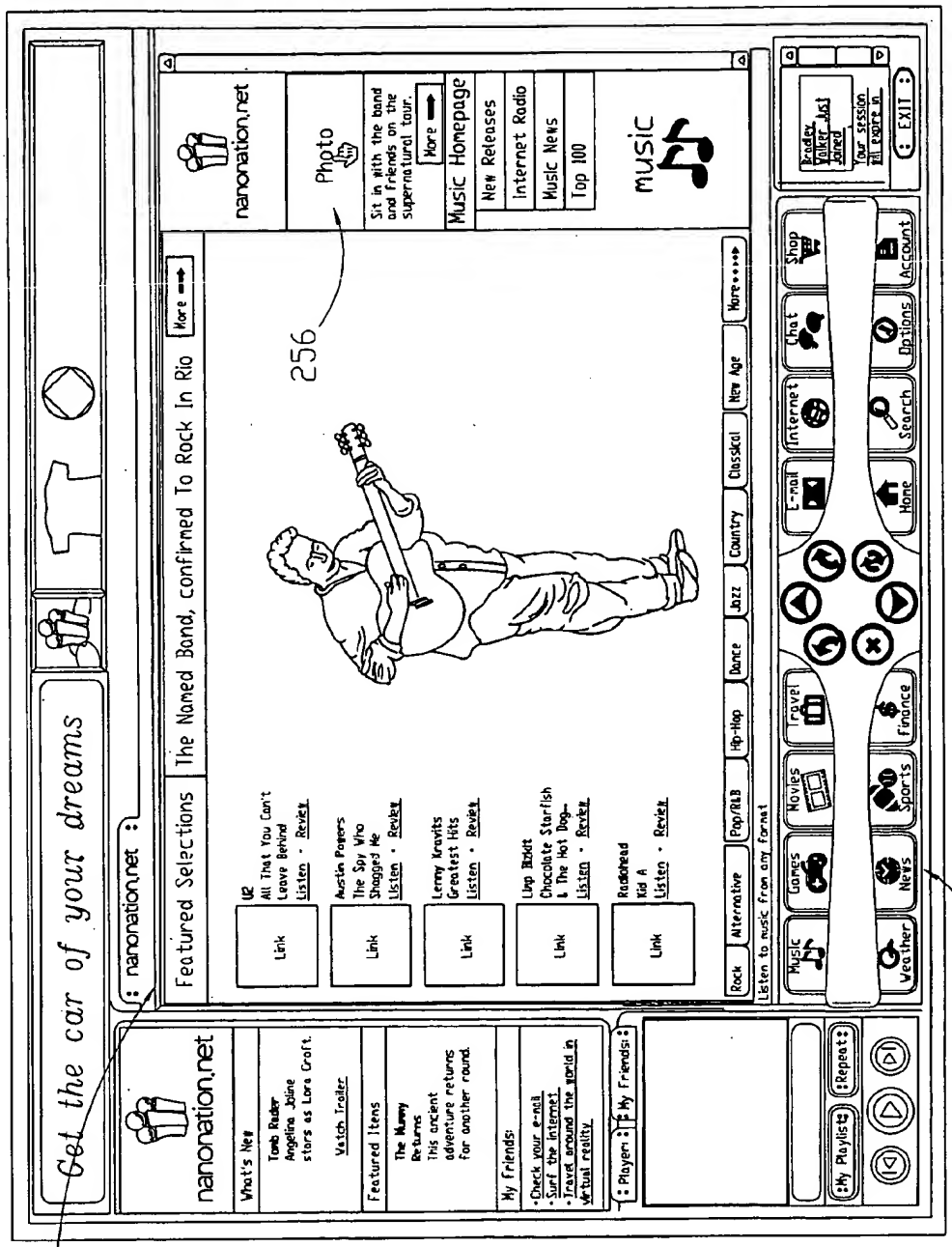

Get the car of your dreams
nanonation.net
Featured Selections
The Named Band, confirmed To Rock In Rio
More
Photo
256
Music Homepage
New Releases
Internet Radio
Music News
Top 100
music
Rock
Alternative
Pop/R&B
Hip-Hop
Dance
Jazz
Country
Classical
New Age
Listen to music from any format
Music
Games
Movies
Travel
News
Sports
finance
Weather
E-mail
Internet
Chat
Shop
Home
Search
Options
Account
EXIT
Player
My Friends
My Playlist
Repeat
258
240

Fig. 11E.

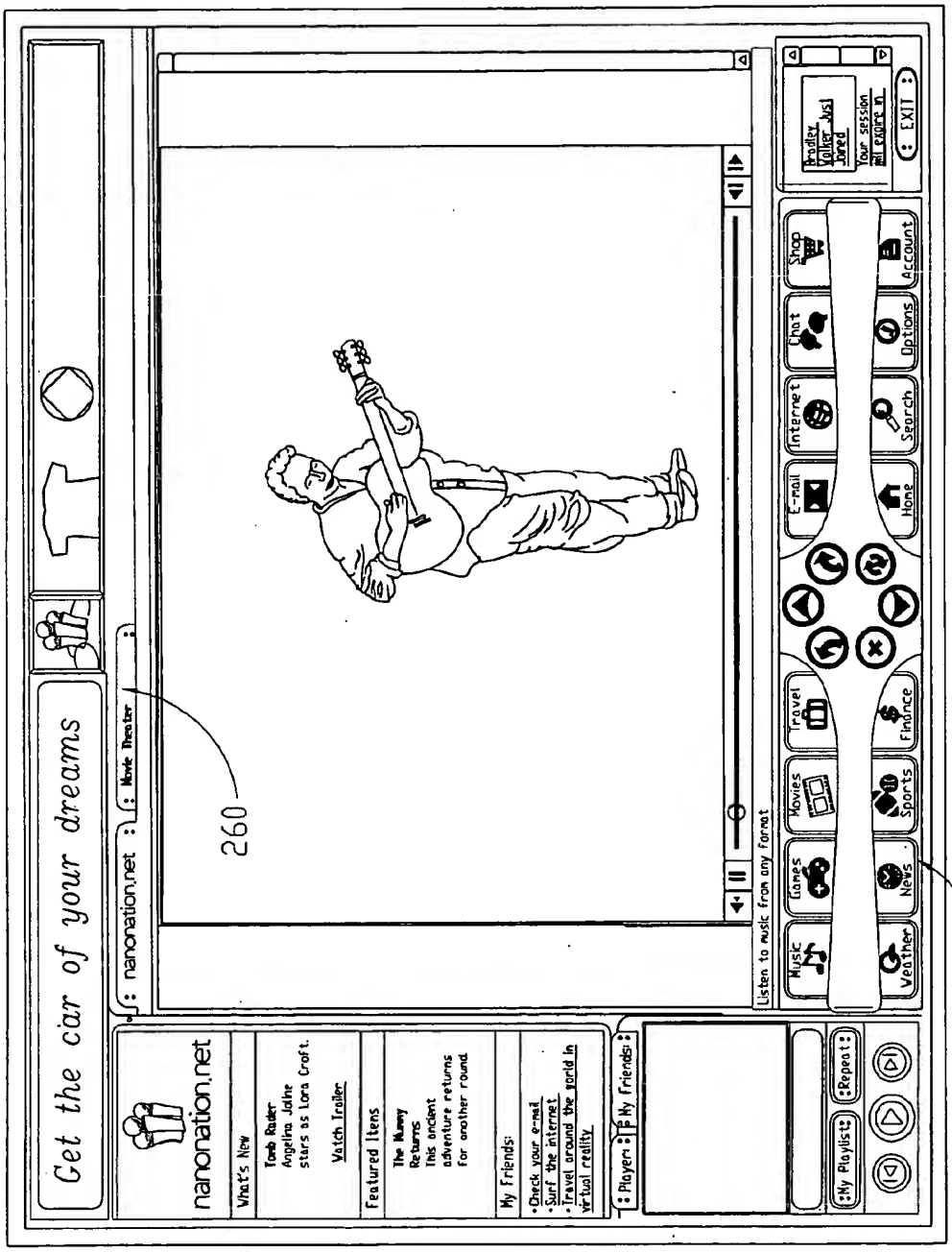

Get the car of your dreams
nanonation.net
What's New
Tomb Raider
Angelina Jolie
stars as Lara Croft.
Watch Trailer
Featured Items
The Mummy
Returns
this ancient
adventure returns
for another round
My Friends:
•Check your e-mail
•Surf the internet
•Travel around the world in
virtual reality
Player
My Friends
My Playlist
Repeat
nanonation.net : Movie Theater
260
Listen to music from any format
Music
Games
Movies
Travel
Weather
News
Sports
Finance
E-mail
Internet
Chat
Shop
Home
Search
Options
Account
EXIT
262

Fig. 11F.

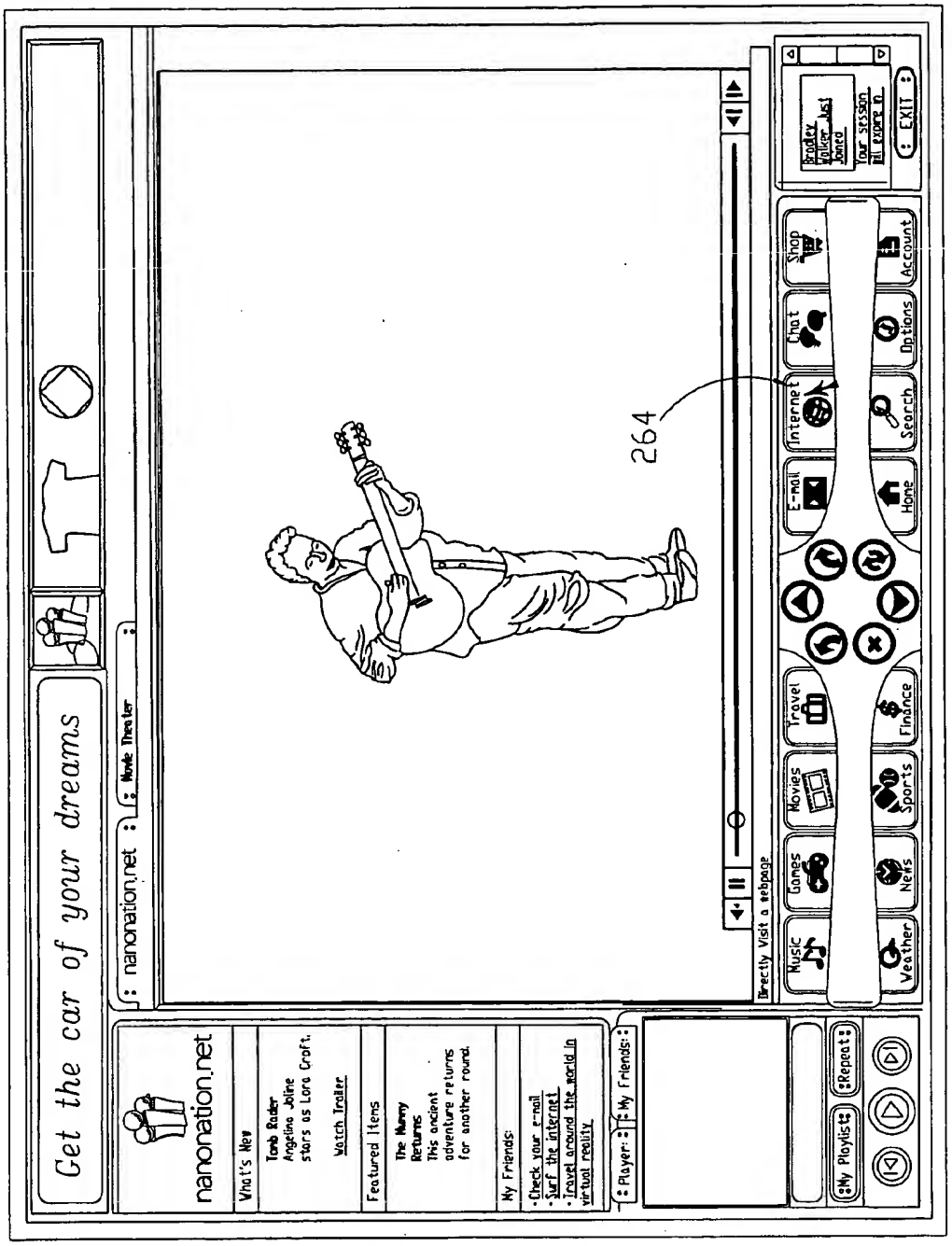

Get the car of your dreams
nanonation.net
What's New
Tomb Raider
Angelina Jolie
stars as Lara Croft.
Watch Trailer
Featured Items
The Mummy
Returns
This ancient
adventure returns
for another round.
My Friends:
·Check your e-mail
·Surf the internet
·Travel around the world in
virtual reality
: Player : : My Friends :
:My Playlist:
:Repeat:
nanonation.net : : Movie Theater :
264
Directly Visit a webpage
Music
Games
Movies
Travel
Weather
News
Sports
Finance
E-mail
Internet
Chat
Shop
Home
Search
Options
Account
: EXIT :

Fig. 11G.

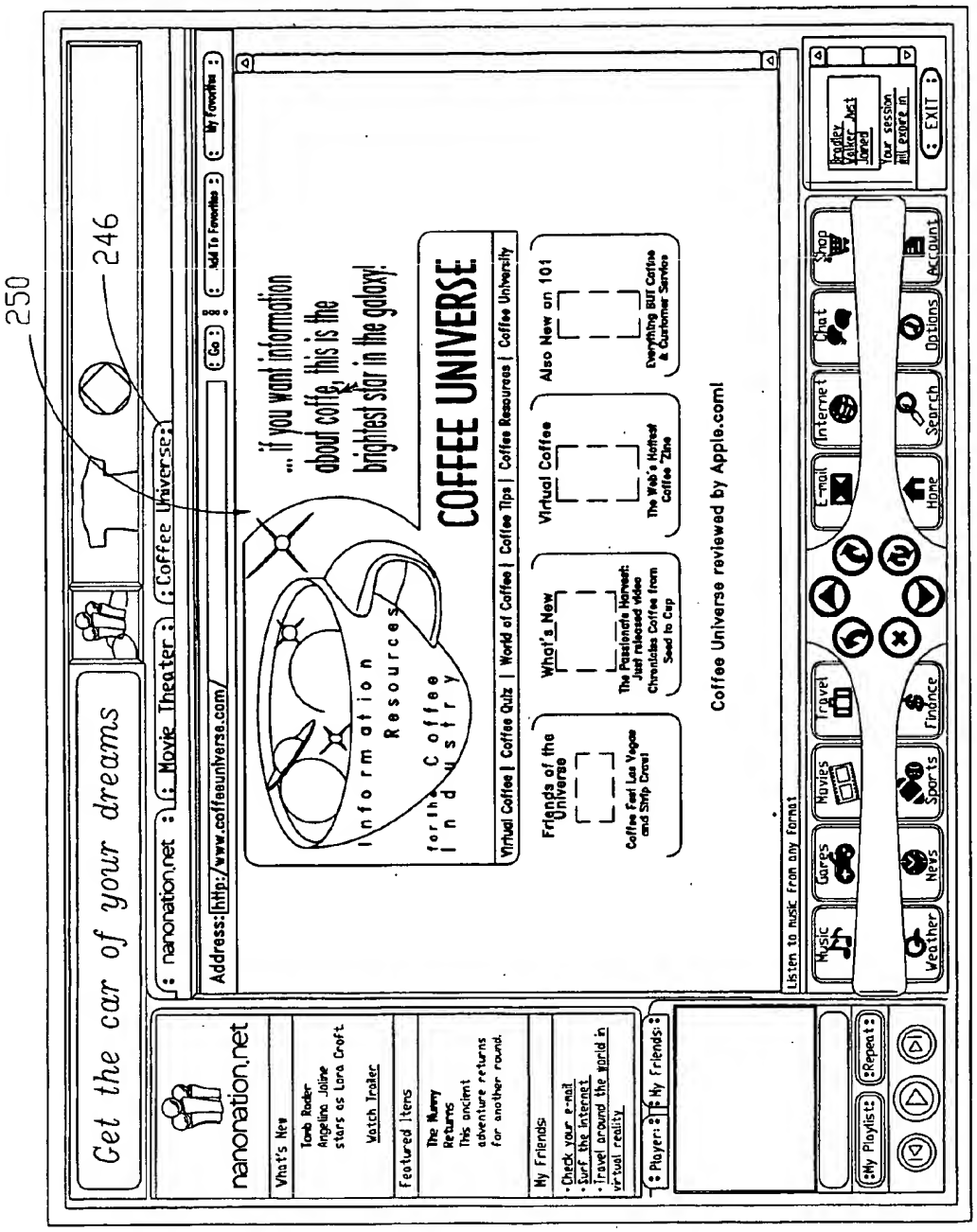

250
246
Get the car of your dreams
nanonation.net
Coffee Universe
Address: http://www.coffeeuniverse.com/
... if you want information about coffee, this is the brightest star in the galaxy!
COFFEE UNIVERSE
Information Resources for the Coffee Industry
Virtual Coffee | Coffee Quiz | World of Coffee | Coffee Tips | Coffee Resources | Coffee University
Friends of the Universe
What's New
Virtual Coffee
Also New on 101
Coffee Fest Las Vegas and Ship Crawl
The Passionate Harvest: Just released video Chronicles Coffee from Seed to Cup
The Web's Hottest Coffee Zine
Everything BUT Coffee & Customer Service
Coffee Universe reviewed by Apple.com!
What's New
Tomb Raider
Angelina Jolie stars as Lara Croft
Watch Trailer
Featured Items
The Mummy Returns
This ancient adventure returns for another round.
My Friends
Check your e-mail
Surf the Internet
Travel around the world in virtual reality
Player:
My Friends
My Playlist
Repeat
Listen to music from any format
Music
Games
Movies
Travel
Sports
Finance
Weather
News
E-mail
Internet
Chat
Shop
Home
Search
Options
Account
EXIT

Fig. 11H.

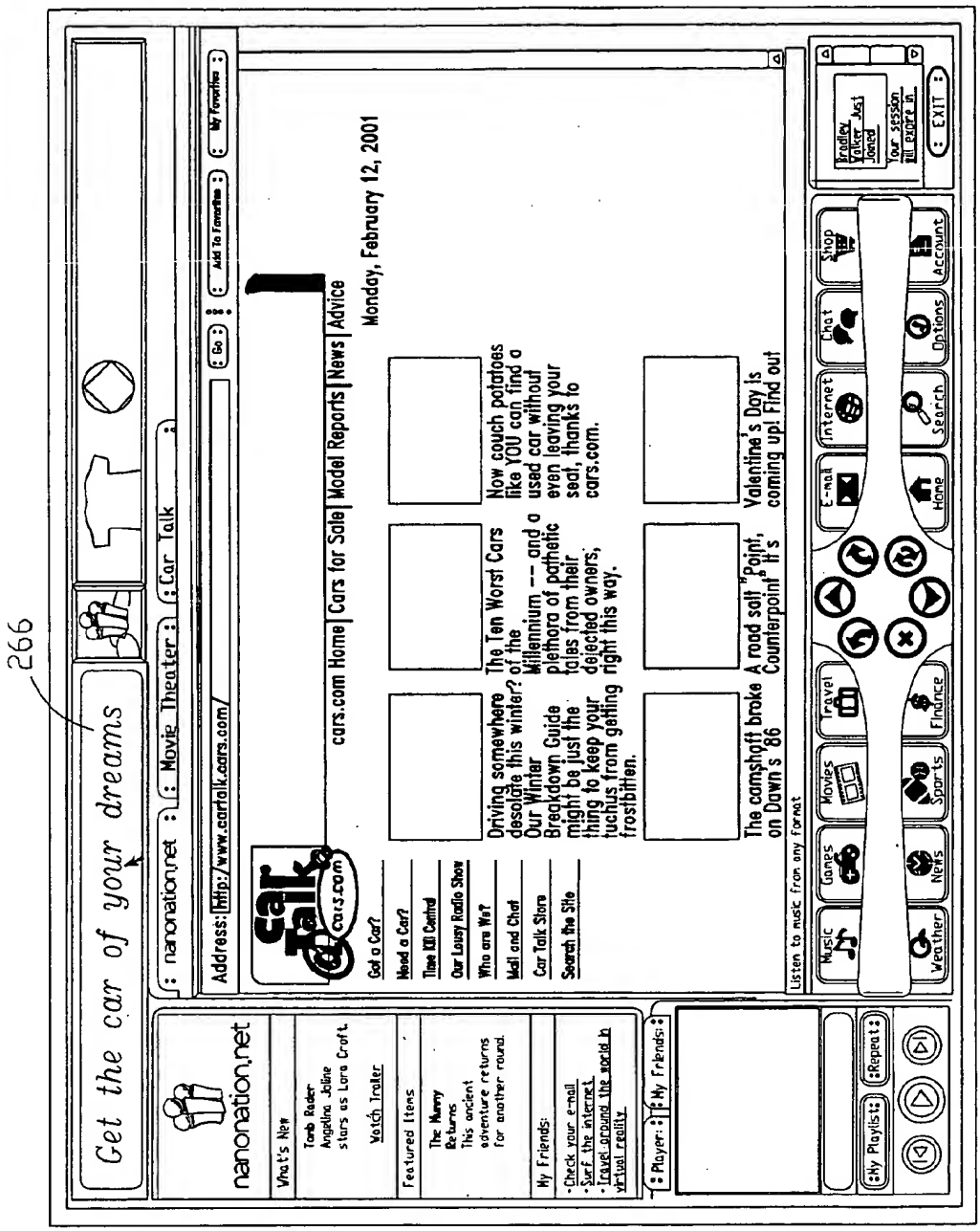

266
Get the car of your dreams
nanonation.net
Address: http://www.cartalk.cars.com/
Car Talk
cars.com
cars.com Home | Cars for Sale | Model Reports | News | Advice
Monday, February 12, 2001
Got a Car?
Need a Car?
Time Kill Central
Our Lousy Radio Show
Who are We?
Mail and Chat
Car Talk Store
Search the Site
Driving somewhere desolate this winter? Our Winter Breakdown Guide might be just the thing to keep your tuchus from getting frostbitten.
The Ten Worst Cars of the Millennium -- and a plethora of pathetic tales from their dejected owners; right this way.
How couch potatoes like YOU can find a used car without even leaving your seat, thanks to cars.com.
The camshaft broke on Dawn's '86
A road salt "Point, Counterpoint" it's
Valentine's Day is coming up! Find out
Listen to music from any format
Music
Games
Movies
Travel
E-mail
Internet
Chat
Shop
Weather
News
Sports
Finance
Home
Search
Options
Account
EXIT
Bradley Baker Just Joined
Your session will expire in
nanonation.net
What's New
Tomb Raider
Angelina Jolie stars as Lara Croft.
Watch Trailer
Featured Items
The Mummy Returns
This ancient adventure returns for another round.
My Friends:
Check your e-mail
Surf the internet
Travel around the world in virtual reality
Player:
My Friends:
My Playlists:
Repeat:

Fig. 111.

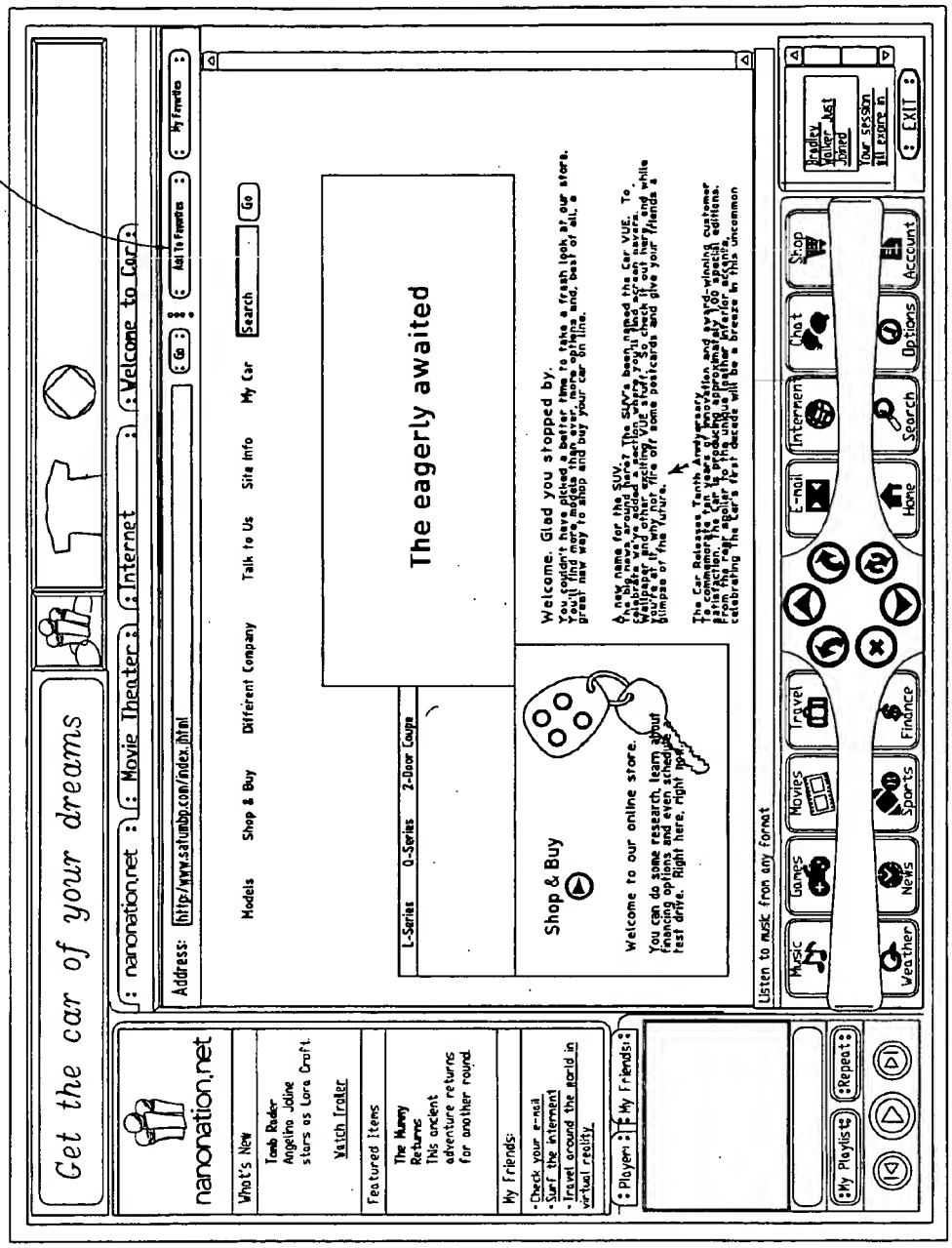

268
Get the car of your dreams
nanonation.net
Welcome to Car
Internet
Movie Theater
nanonation.net
Address: http://www.saturnbbs.com/index.html
Models
Shop & Buy
Different Company
Talk to Us
Site Info
My Car
Search
L-Series
Q-Series
2-Door Coupe
The eagerly awaited
Shop & Buy
Welcome. Glad you stopped by.
Welcome to our online store.
What's New
Tomb Raider
Angelina Jolie
stars as Lara Croft
Watch Trailer
Featured Items
The Mummy
Returns
This ancient
adventure returns
for another round
My Friends:
Check your e-mail
Surf the Internet!
Travel around the world in
virtual reality
Player
My Friends
My Playlist
Repeat
Listen to music from any format
Music
Games
Movies
Travel
Weather
News
Sports
Finance
E-mail
Internet
Chat
Shop
Home
Search
Options
Account
EXIT

Fig. 11J.

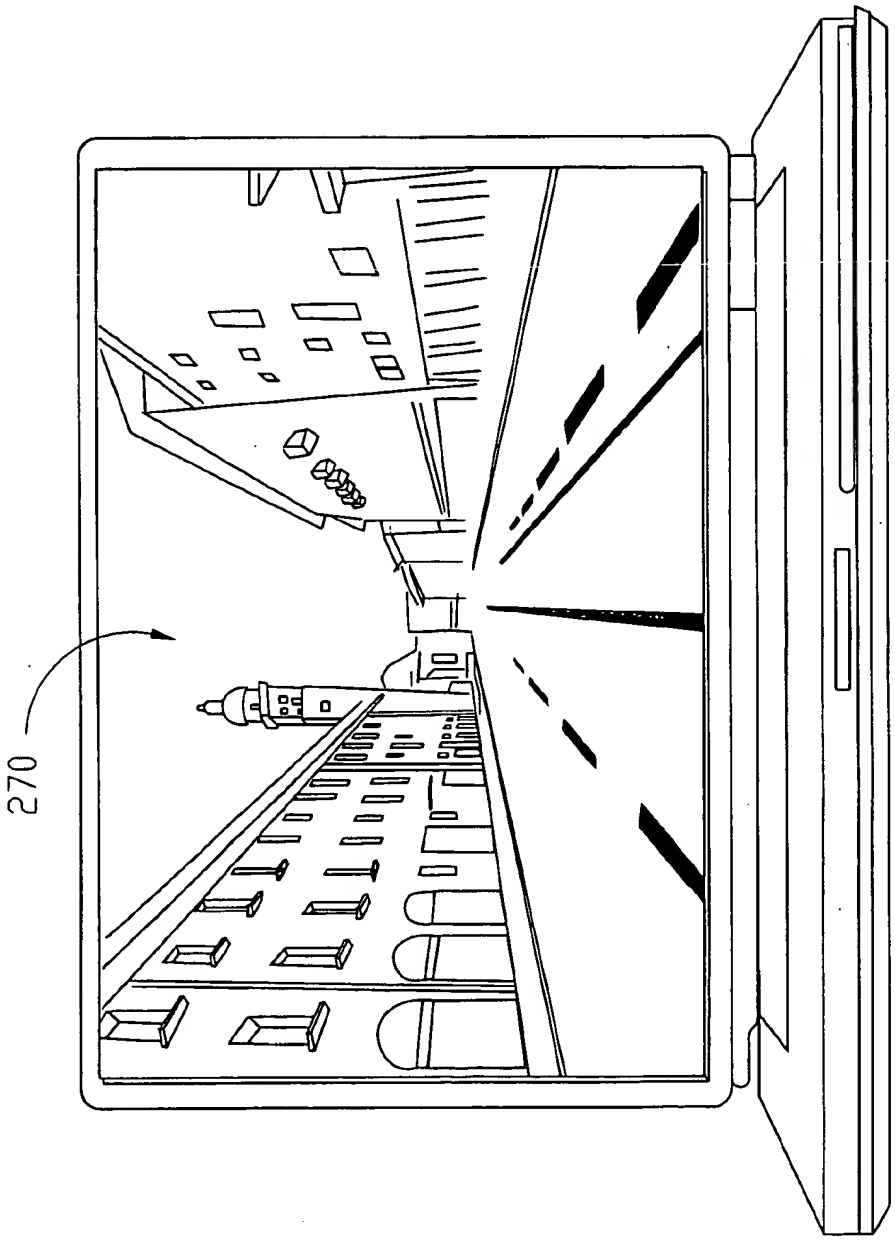


Fig. 12A.

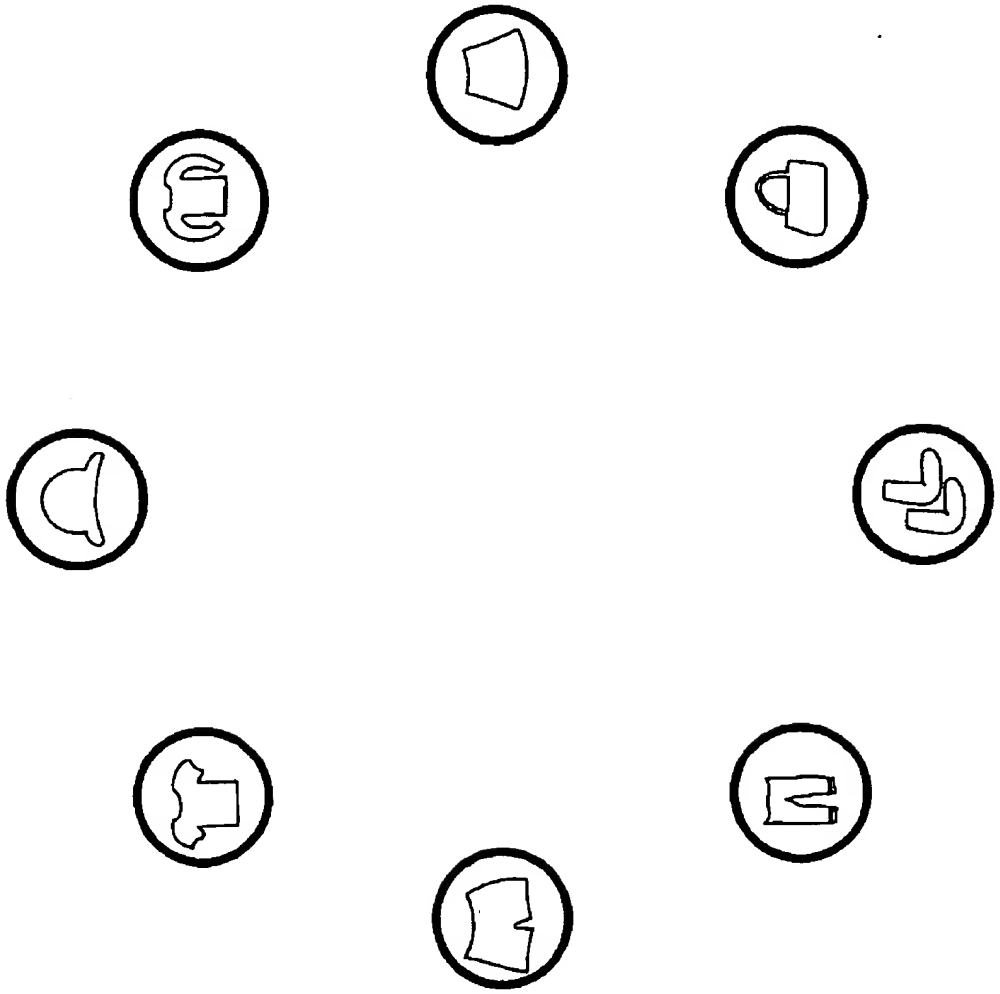


Fig. 12B.

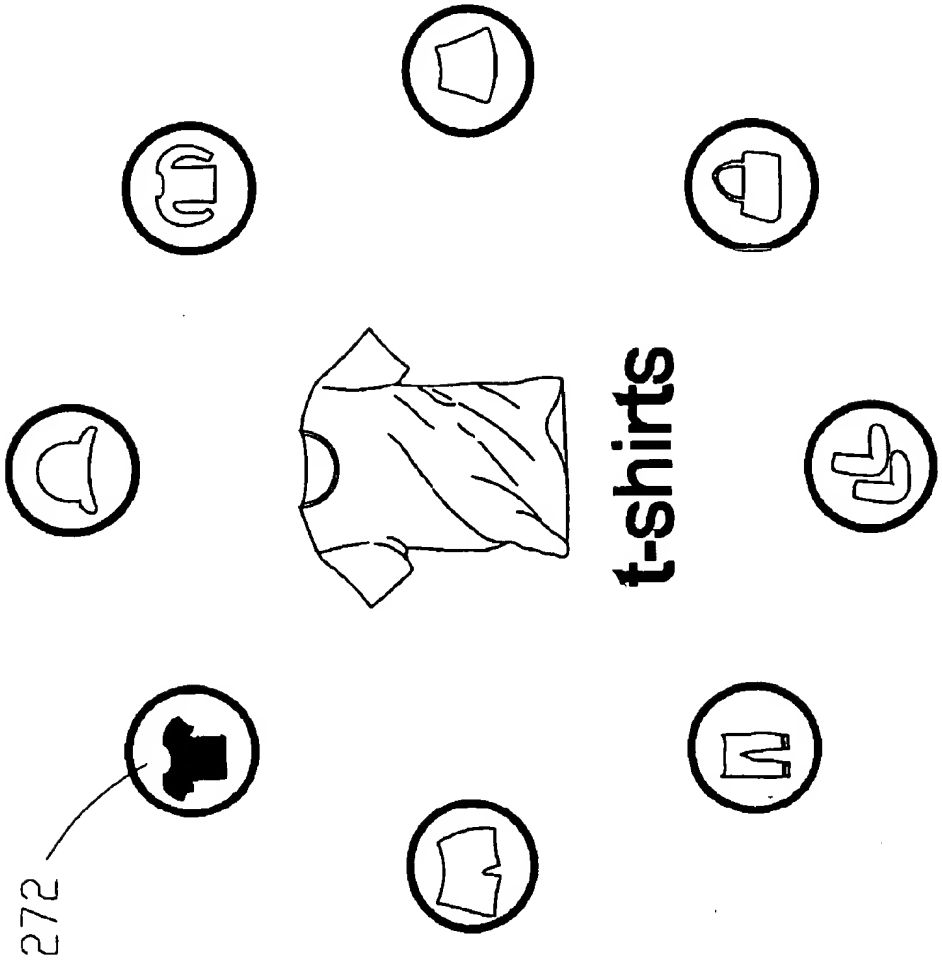


Fig. 12C.

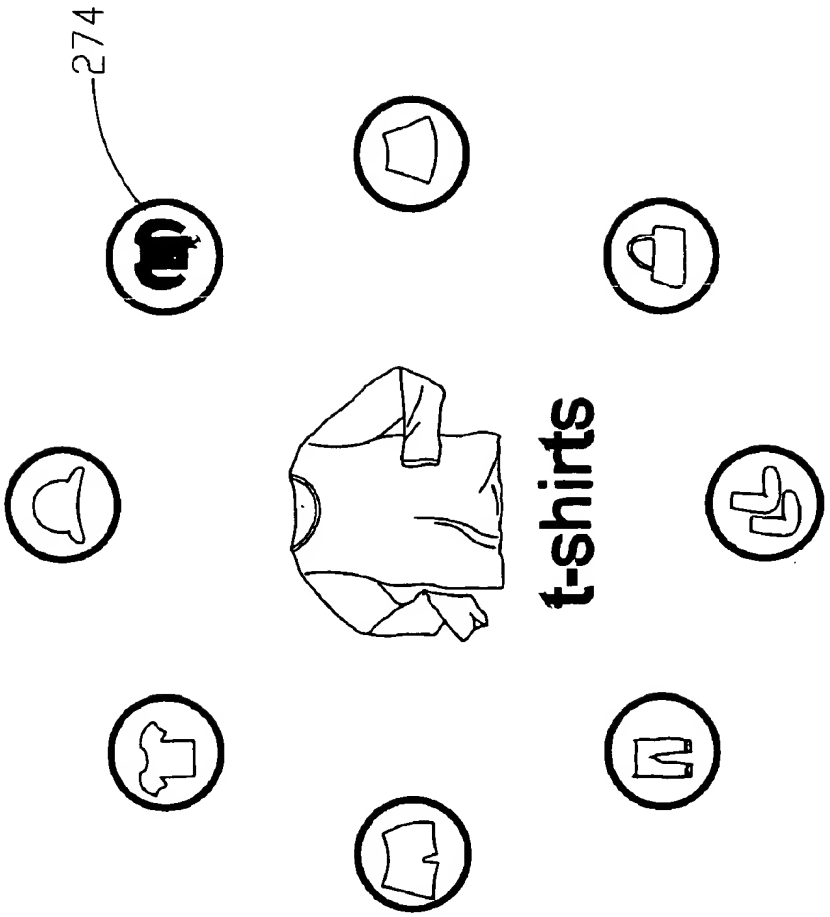


Fig. 12D.

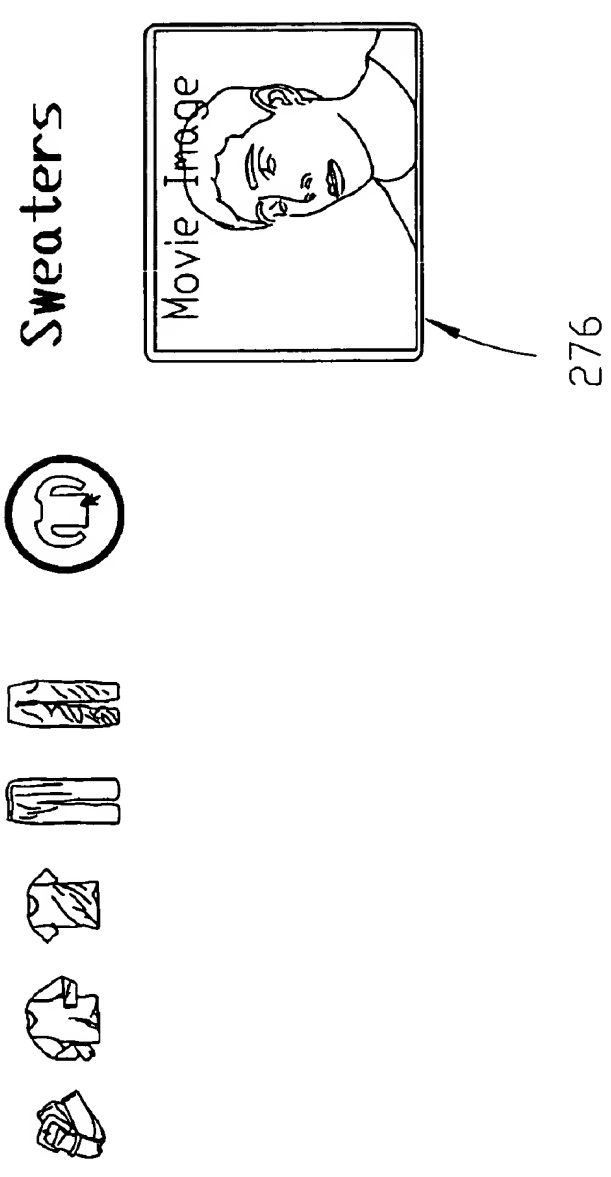


Fig. 12E.

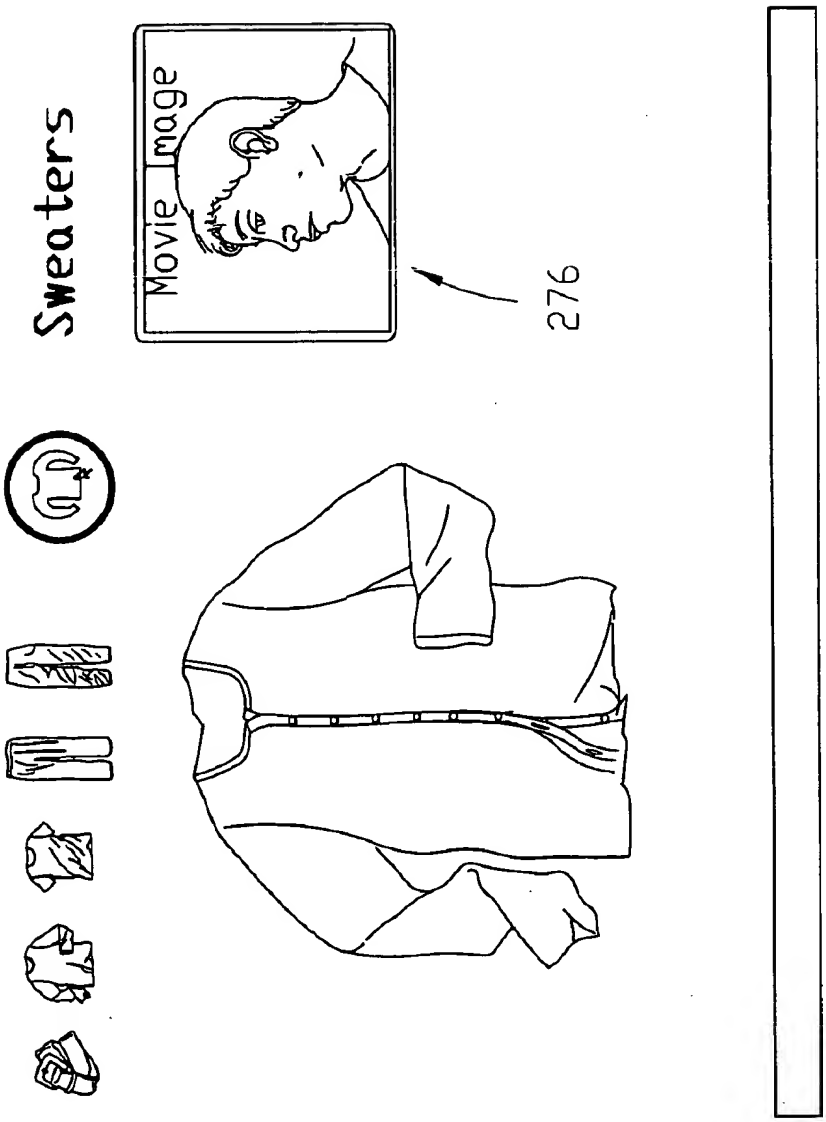


Fig. 12F.

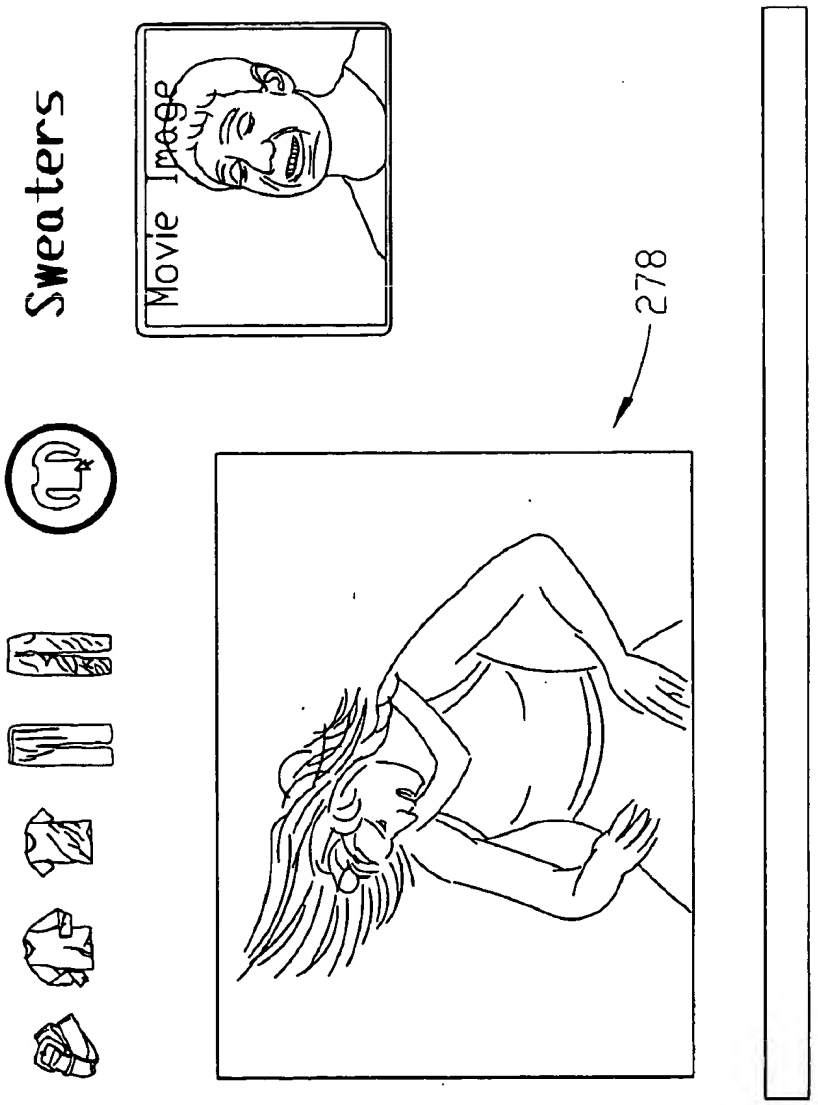


Fig. 12C.

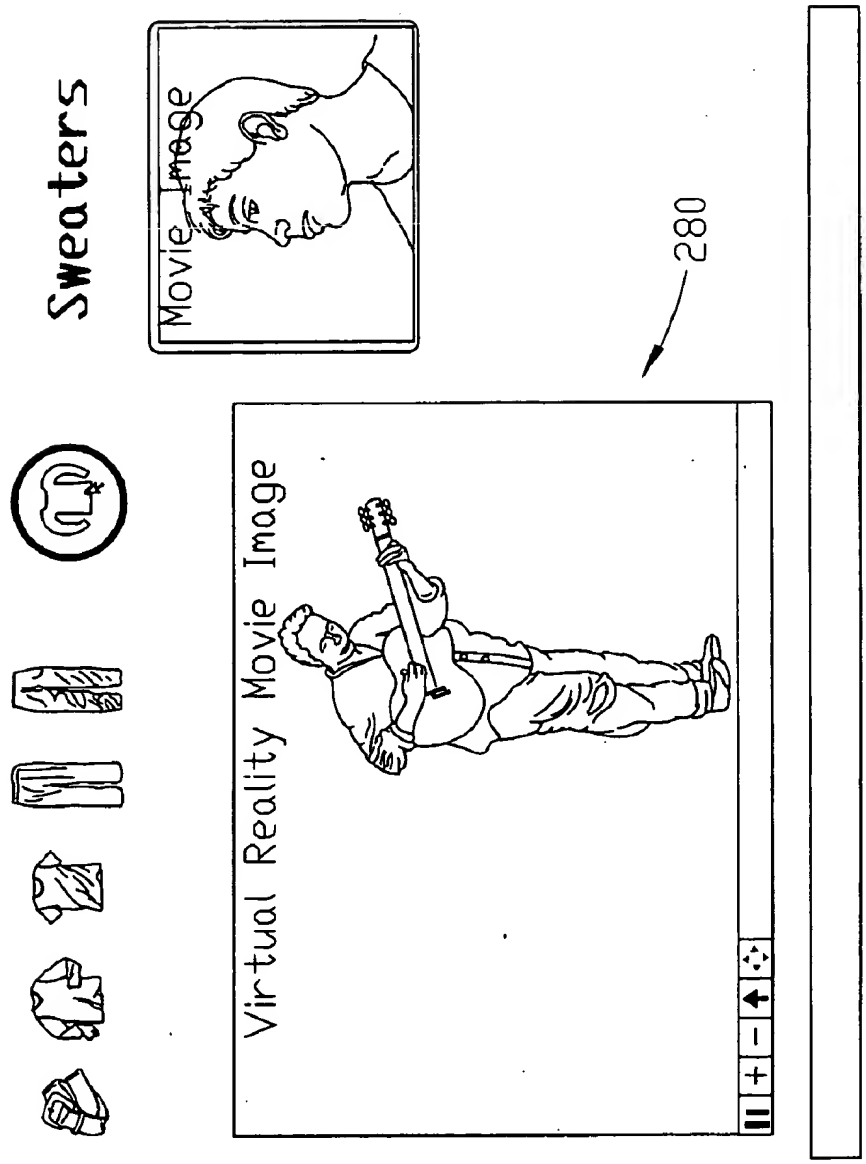


Fig. 12H.

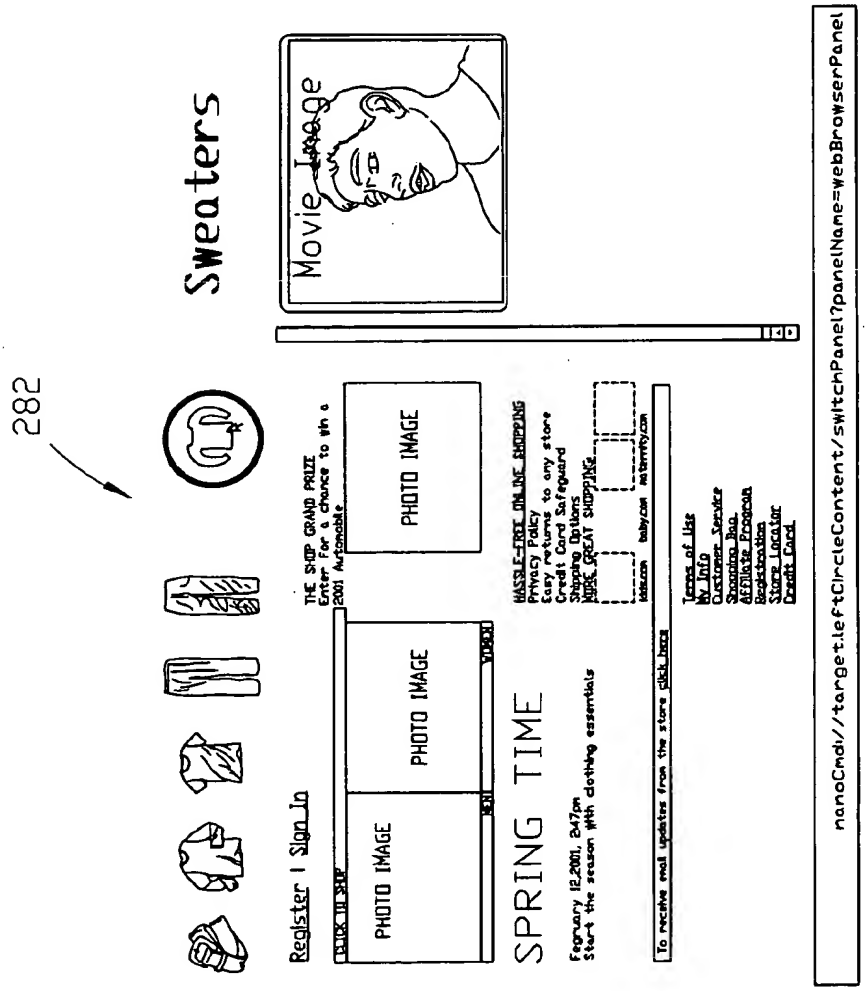

282
Sweaters
Movie Image
Register | Sign In
THE SHOP GRAND PRIZE
Enter for a chance to win a
2001 Automobile
PHOTO IMAGE
PHOTO IMAGE
SPRING TIME
HASSLE-FREE ONLINE SHOPPING
Privacy Policy
Easy returns to any store
Credit Card Safeguard
Shipping Options
MORE GREAT SHOPPING
February 12,2001, 2:47pm
Start the season with clothing essentials
To receive email updates from the store click here
Terms of Use
My Info
Customer Service
Shopping Bag
Affiliate Program
Registration
Store Locator
Credit Card
nanoCmd://target.leftCircleContent/switchPanel?panelName=webBrowserPanel

Fig. 121.

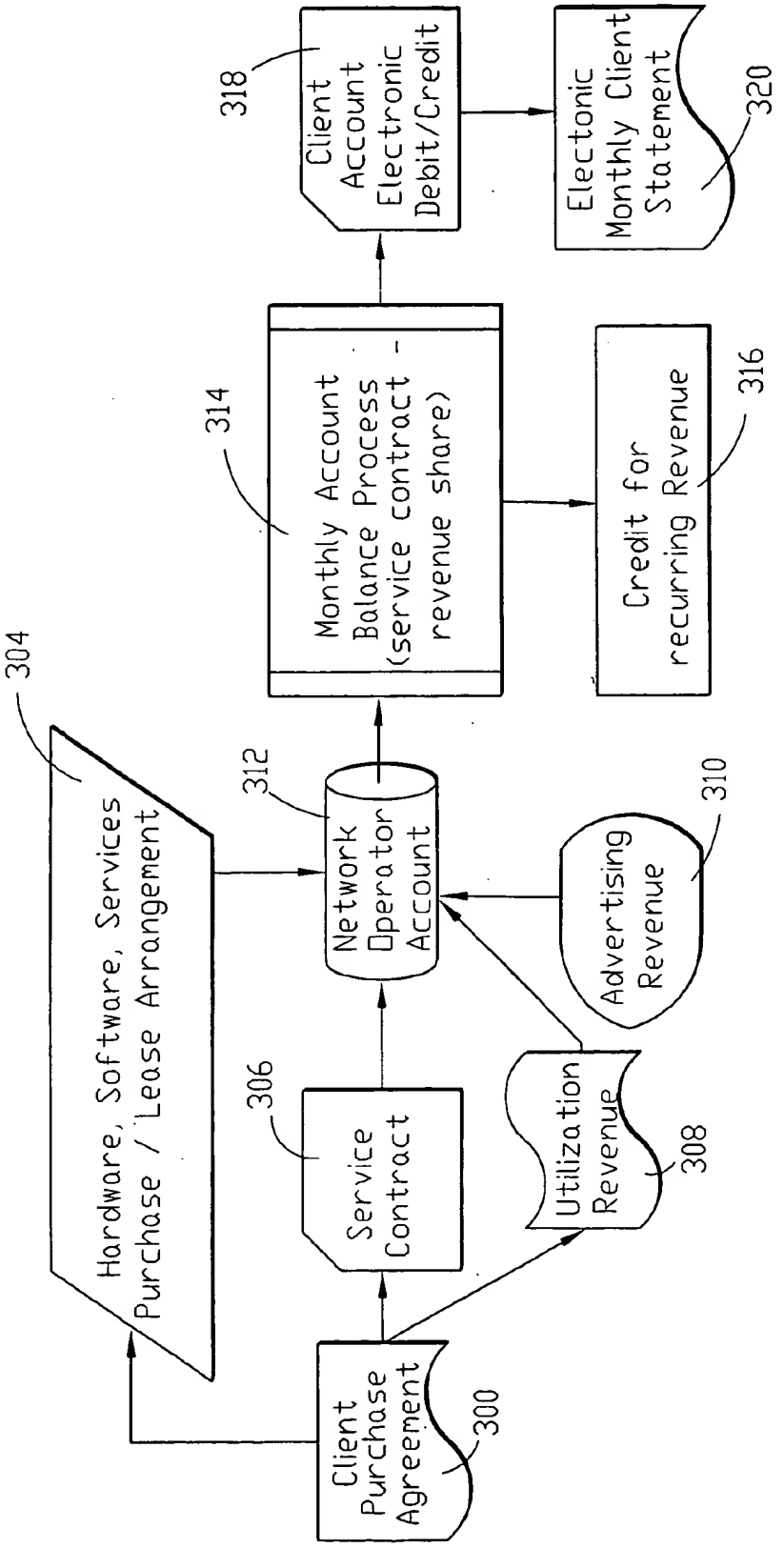

Client Purchase Agreement
300
Hardware, Software, Services Purchase / Lease Arrangement
304
Service Contract
306
Utilization Revenue
308
Advertising Revenue
310
Network Operator Account
312
Monthly Account Balance Process (service contract - revenue share)
314
Credit for recurring Revenue
316
Client Account Electronic Debit/Credit
318
Electonic Monthly Client Statement
320

Fig. 13.

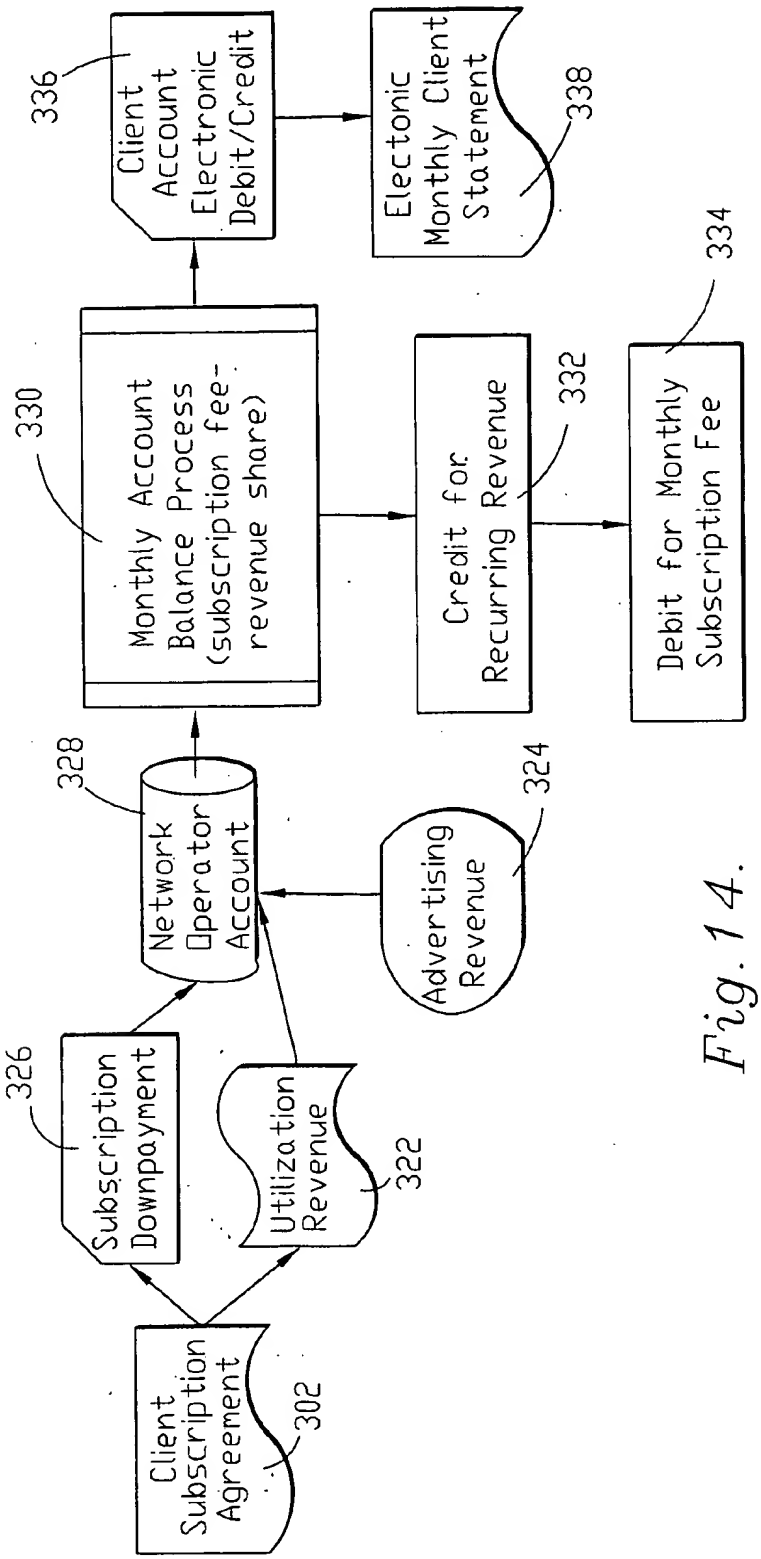

Client Subscription Agreement
302
Subscription Downpayment
326
Utilization Revenue
322
Advertising Revenue
324
Network Operator Account
328
Monthly Account Balance Process (subscription fee-revenue share)
330
Credit for Recurring Revenue
332
Debit for Monthly Subscription Fee
334
Client Account Electronic Debit/Credit
336
Electonic Monthly Client Statement
338

Fig. 14.

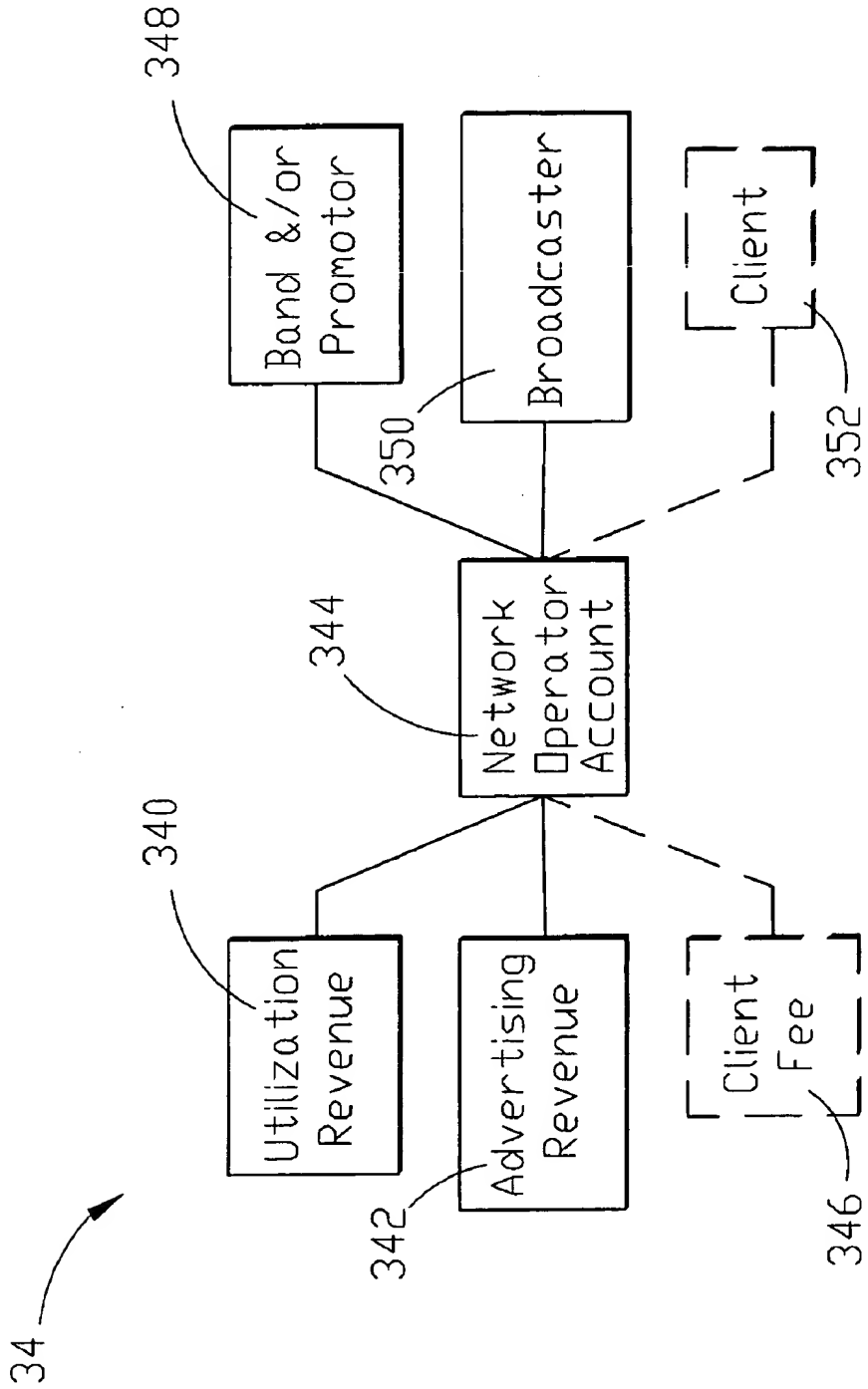

348
Band &/or Promotor
Broadcaster
Client
350
352
344
Network Operator Account
340
Utilization Revenue
Advertising Revenue
Client Fee
342
346
34

Fig. 15.

COMPUTER NETWORK HAVING CONTEXT SENSITIVE APPLICATIONS AND CONTROLS FORMING DYNAMIC USER INTERFACES ON LOCAL COMPUTER TERMINALS

COPYRIGHT NOTICE AND AUTHORIZATION

[0001] A portion of the disclosure of this patent document contains material, which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever.

FIELD OF THE INVENTION

[0002] This invention relates to computer networks providing adaptive and context sensitive interactive multimedia applications and controls. The applications and controls combine to form dynamic user interfaces on local computer terminals. More particularly, the invention relates to adaptive computer networks having centrally located servers receiving context and user information from local computer terminals to develop, utilizing extensible markup language (XML) packets, a context and user sensitive interface for the local computer terminals. The interfaces are created with centrally stored files, locally stored files, and/or Internet retrievable files thus providing centrally programmable user interfaces for local computer terminals.

BACKGROUND OF THE INVENTION

[0003] It is rumored that several decades ago, a Harvard professor opined that the world would never need more than three or four computers. In contrast to this legendary lack of foresight, an overwhelming majority of businesses and homes utilize personal computers and computer networks privately for word processing and computational and database support. However, computer use has not expanded as quickly into arenas where the general public uses them. Generally, computer systems have lacked the versatility, durability, and reparability for wide spread use by the general public.

[0004] Computers have been utilized in public facilities such as libraries for many years where individuals from the general public use the computers for very limited purposes and the presence of food and drink is restricted. Computers have also been used in restaurants, bars, and other venues to provide limited entertainment. By limiting the number and complexity of controls and enclosing the components of the computers in single housings, such as an inlay table or conventional standup arcade housing, these devices have proven sufficiently durable for the general public to use. However, it is generally necessary to remove the entire computer and housing from the venue in order to perform significant repairs.

[0005] Typically, entertainment is provided in the form of a jukebox, which only plays music, or it is provided in the form of a single game device, on which a user plays a single video game. Recently, some game devices have been enhanced to provide multiple game selections. The multiple player features of these video games, with the exception of recently introduced trivia games, are not capable of interactive play between two different players on different computers. Typically, each competing player takes turns or they play on a single computer having multiple sets of controls.

[0006] The recent proliferation of the Internet has led to a small increase of installations in libraries and other public forums such as coffee shops allowing individuals to conduct on-line research or browse while enjoying a cup of coffee. However, even with the dramatic increases in computer processor speed and memory capacity, which have significantly enhanced the computer's capability to support media applications such as videos, music, and interactive gaming, no computer network has provided sufficient versatility and adaptability for wide spread deployment in public venues such as bars, restaurants, hotels, and airports. While pay-per-view and opt-in satellite channels have started to deliver more content options, they have not utilized the web or created an avenue for easily uploading original content from client locations

[0007] In these environments, customer interests and time variables for example, change dramatically from one venue to another and from one user to another. Current networks in these environments are custom designed for each specific venue and lack the ability to adapt to different customer interests and desires and generally lack the ability to provide more than one type of media at a time. Further the screen displays or graphical user interfaces (GUI's) for these applications are difficult to change. When a change is desired, the new GUI must be programmed and stored locally on the computer terminals of the network. Other solutions utilize Internet resources and link to customized web sites created and sponsored by the venue owner. These web-based implementations use standard browser technology utilizing the entire screen for the browser and thus fail to provide a true multimedia solution.

[0008] For years, single-purpose computer platforms or kiosk systems have been displaying information to public environments. Initially, these systems featured hard-coded presentation applications, which ran in a perpetual loop on the system. The user's options were limited to pre-programmed functions and/or paths through the content provided. More recently, some of these systems have included web-based components, but they have still failed to incorporate digital movie capabilities, allow for multiple application tasks within the user interface, or allow for the dynamic management of local resources.

BRIEF SUMMARY OF THE INVENTION

[0009] There is, therefore, provided the practice of the invention, a context-sensitive user interface generated from a central location for display and use on remote terminals. The user interface broadly includes a plurality of screen application regions and an interactive movie including control elements. The interactive movie is displayed in a selected one of the plurality of screen application regions.

[0010] In a preferred embodiment, the control elements comprise application tabs and control buttons. Different sets of control buttons are provided to control different applications including browsers and video. As the user selects different applications the control movie is changed to display a different set of control buttons.

[0011] In another aspect of the present invention, the user interface includes a plurality of screen application regions,

and a background movie played beneath substantially all of the screen application regions. A browser operates in a selected one of the screen application regions and overlays and is embedded in the background movie.

[0012] In a preferred embodiment, the browser is controlled by the interactive control movie. Further, it is possible to open additional applications in the selected one of the screen application regions. Identification tabs are displayed for the browser and other application panels for easy access by the user. Preferably, the identification tabs are provided by an interactive movie.

[0013] In still another aspect of present invention a dynamic user interface is provided having a plurality of screen application regions and multiple applications operating in a selected one of the screen application regions. In a preferred embodiment identification tabs are provided for each of the application panels. Preferably, at least one of the applications is a browser.

[0014] Each of the above-described aspects of the user interface are utilized in methods for providing, generating, and controlling the user interface. Further, the dynamic user interface and methods operate on computer networks including a plurality of remote terminals in operative communication with a central server. The remote terminals include displays operative to display the above-described aspects of the user interface.

[0015] In a still further aspect of the present invention, a computer installation is provided for installing the remote terminals in a wall. The installation comprises at least one wall frame member and a wall cover member attached to the wall frame member. A computer housing, which supports a central processing unit, is mounted on the wall frame member adjacent to an inner side of the wall cover member. A computer display in operative communication with the central processing unit is positioned adjacent to an outer side of the wall cover member, and an input device is located for access by a user. The input device is also in operative communication with the central processing unit.

[0016] In a still further aspect of the present invention, a method is provided for distributing income from advertisers and from transmission of a media event. The method includes a network operator receiving income from advertisers and from customers viewing the media event at various venues. The network operator retains an operator portion of the income and distributes a promotor portion of the income to the promotor. Further, the network operator distributes a venue portion of the income to the owner of the venue.

[0017] Accordingly, it is an objective of the present invention to provide an improved computer network with an improved user interface for controlling multiple media applications.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] These and other inventive features, advantages, and objects will appear from the following Detailed Description when considered in connection with the accompanying drawings in which similar reference characters denote similar elements throughout the several views and wherein:

[0019] **FIG. 1** is a schematic illustration of a computer network according to the present invention;

[0020] **FIG. 2** is a perspective view of a computer terminal provided in the computer network of **FIG. 1**;

[0021] **FIG. 3** is a schematic block diagram illustrating software and data components of the computer terminal in **FIG. 2**;

[0022] **FIG. 4** is a side view and partial cross section of an alternate computer terminal installation according to the present invention;

[0023] **FIG. 5** is a fragmentary rear view of the computer terminal installation in **FIG. 4**;

[0024] **FIG. 6** is a schematic device diagram illustrating the various communication means for the computer network according to the present invention;

[0025] **FIG. 7** is a schematic block diagram illustrating various hardware, software, and data components utilized in the computer network of **FIG. 1**;

[0026] **FIG. 8** is a block diagram illustrating the steps for creation of an overall user interface according to the present invention;

[0027] **FIG. 9** is a block diagram illustrating the steps and updating or modifying the overall user interface;

[0028] **FIG. 10** is a schematic block diagram illustrating software, hardware, and data components of the computer network of **FIG. 1**;

[0029] **FIG. 11A** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0030] **FIG. 11B** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0031] **FIG. 11C** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0032] **FIG. 11D** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0033] **FIG. 11E** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0034] **FIG. 11F** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0035] **FIG. 11G** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0036] **FIG. 11H** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0037] **FIG. 11I** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0038] **FIG. 11J** is an elevational view of a user interface illustrating a particular step in a user entertainment session;

[0039] **FIG. 12A** is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0040] **FIG. 12B** is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0041] **FIG. 12C** is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0042] **FIG. 12D** is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0043] FIG. 12E is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0044] FIG. 12F is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0045] FIG. 12G is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0046] FIG. 12H is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0047] FIG. 12I is an elevational view of a user interface illustrating a specific step in a user commercial session;

[0048] FIG. 13 is a schematic block diagram of a client purchase agreement business plan;

[0049] FIG. 14 is a schematic block diagram of a client subscription agreement business plan; and

[0050] FIG. 15 is a schematic block diagram of a revenue division model according to the present invention.

DETAILED DESCRIPTION

[0051] Referring to the drawings in greater detail, FIG. 1 shows a computer network 20 constructed in accordance with a preferred embodiment of the present invention. The computer network 20 broadly includes a local installation 21 having a plurality of local terminals 22 and an optional local server 24. The computer network also includes several central servers 26, 28, 30. The local terminals 22 and the local server 24 reside at the physical location of a forum such as a sports bar. As illustrated in the schematic block diagrams of FIGS. 7 and 10, the computer network 20 is operative to provide a dynamic, centrally programmed, GUI (graphical user interface) 32 shown in FIG. 2. The computer network 20 also enables the use of a unique income distribution model 34 illustrated in FIG. 15.

[0052] Referring to FIG. 2, the local terminals 22 each preferably includes a display 36, input device 38, and a schematically illustrated CPU (central processing unit) 40 together with a CPU housing 42 and necessary OS (operating system) software. Thus, each terminal 22 is preferably a complete computer system. The specific platform of the terminals is not critical, can change from terminal to terminal, and may include Windows or Macintosh platforms, for example. The display 36 is in operative communication with the CPU 40 for control by the CPU 40, and the display 36 includes a frame 44, which supports speakers within speaker openings 46 and a centrally located and recessed headphone jack 48. While the position of the speakers and sound jack may vary, the speakers with their accompanying speaker openings 46 and the sound jack 48 are preferably positioned adjacent the bottom edge 49 of the display frame 44. Generally the speakers will be positioned at the base of the monitor, but they may be placed in the CPU housing 42 in some configurations. The sound jack 48 can also be positioned on the input device, preferably on the right side of the keyboard tray in the embodiment shown. The display 36 can be a high resolution CRT (cathode ray tube) monitor such as a SVGA (super video graphics array) capable monitor. Preferably, an LCD (liquid crystal display) monitor or an FTM (flat technology monitor) having a digital connection is utilized.

[0053] The input device 38 preferably includes multiple input components. In the embodiment shown, the input device 38 utilizes a moisture and impact resistant keyboard 50 having illuminated or back lit keys. The illuminated keys permit use in dimly lit areas such as bar room environments, and the durability of the keyboard resists the liquid spills and bumps that are expected in such venues. A second component of the input device comprises a relative pointer such as the illustrated track ball 52 and thumb click button 54. Both the track ball 52 and click button 54 are preferably sealed to resist moisture penetration. Alternatively, a mouse is utilized in place of the track ball 52 or an absolute pointer such as a touch screen is provided. The track ball 52 and click button 54 are preferred because they are held by an input device housing 56 which also holds the keyboard 50. Including all components of the input device 38 in the single input device housing 56 minimizes the likelihood of damage and theft. In one embodiment, the input device is in remote communication, preferably through infrared signals, with the CPU 40.

[0054] The CPU 40 is held inside the CPU housing 42 along with other necessary components of the terminal, for example, SCSI (small computer system interface) or IDE controller, storage (hard drive), graphics card, communication device/network connection, memory (at least 128M RAM), power supply, and cooling device. The microprocessor should be sufficiently fast for network operations and preferably 700 MHz or greater, while processors operating at approximately 450 MHz are satisfactory. Similarly, the other components are desirably advanced to operate graphic intensive applications and games. In one embodiment, the CPU 40 and other computer system components are provided in a module. The module can be quickly removed from the CPU housing 42 and replaced with another module. This minimizes down time for malfunctioning terminals, and permits maintenance of malfunction CPU's at central repair locations.

[0055] Referring to FIG. 3, users 43 enter information at the terminals 22 using the input devices 38 (FIG. 2). The full-screen GUI 32 developed, as described below, with the client application software 45 facilitates entry of the user information. Users 43 enter, for example, payment information, personal information such as name, gender, and age, and personal preferences such as hobbies, favorite athletic teams, and their alma mater. If desired, the terminals can be provided with magnetic card readers for credit card or prepaid card payment. This user information is stored by the database server 30, described in greater detail below, and may also be stored in local data files 57. During terminal 22 operation, the client application logic 45 utilizes an OS (operating system) 59. The OS is a software component operable to manage network calls, onscreen draw commands, local file storage, and access to physical devices controlled by the CPU. The OS also manages all external applications operating on the same CPU and all hardware connected to the CPU. Thus, the OS runs and manages operation of the client application software 45 which in turn manages and draw resources from data tables 61, which are cached in memory and contain, for example, user information, the local data files 57, and local media files 63. To further enable development of the GUI 32, the client application software 45 is also operable to draw on resources from the network services 65, the network servers 26, 28, 30 (FIG. 1) and other remote data 67 from, for example, Internet sites as managed by the network servers. Further, the client application software 45 inhibits user 43 access to the file systems and OS 59.

[0056] Referring to **FIGS. 4 and 5**, an alternate wall mounted terminal **60** utilizes a unique computer installation to save space. The wall-mounted terminal **60** includes a CPU **62** supported by and preferably held inside a CPU housing **64**. The CPU housing **64** is mounted on at least one substantially rigid wall frame member **66** but preferably on two wall frame members **66, 68** with mounting brackets **67**. Preferably, the housing **64** is mounted between two wall frame members **66, 68** and adjacent an inner side **70** of a substantially constant wall cover member **72** attached to the wall frame members **66, 68**, so that the inner side **70** abuts the wall frame members **66, 68**. A digital computer display **74** is positioned adjacent an outer side **76** of the wall cover member **72** along with speakers **78**, which can be integrated flat speakers mounted in a frame surrounding the display **74**. The display is in operative communication with the CPU for control by the CPU **62**. Other devices, such as an infrared sensor **80** or camera **82** are mounted on the frame or the display adjacent the outer side **76** of the wall cover member **72**. The infrared sensor **80** establishes operative remote communication between the input device **38**, which is made accessible to the user, and the CPU, and users utilize the camera **82** for live video communication. If desired, a second or rear wall cover is attached over the CPU housing **64** and is provided with an access panel.

[0057] The display **74** can be mounted to a selected one of the wall cover member **72**, the wall frame members **66, 68**, and/or the CPU housing **64**. In a preferred embodiment, the display **74** is mounted to the CPU housing utilizing at least one hollow tube **84** passing through the wall cover member **72** and used to convey communication lines and power supply lines between the CPU **62** and the display **74** and speakers **78**. The CPU **62** and other components of the local terminal computer system are supplied with power from a power supply **86** inside the wall cover member **72**. Again, the CPU **62** and other components can be provided in an interchangeable module.

[0058] Referring again to **FIG. 1**, the local server **24** is in operative communication with the terminals **22** through network connections, which may be wired or wireless. The local server **24**, which is not included in all local network installations, assists in system and network operations including file transfers, cache management, application service, and media access allocation. The local server **24** or the terminals **22** if there is no local server, preferably connect to the central servers **26, 28, 30** through the Internet **90**, and the Internet connection is established through commercially available telecommunication services. While an Internet connection is preferred, it is understood that the computer network could be implemented, for example, over a LAN (local area network), WAN (wide area network), intranet, or VPN (virtual private network).

[0059] The central servers **26, 28, 30** include a web server **26**, business server **28**, and database server **30**. The hardware supporting the web server **26** also supports a SOAP (simple object access protocol) server. Each of the servers comprises a commercial server product designed, configured, and/or programmed to perform the desired functions. The web server **26** preferably utilizes Apache Tomcat. The business server **28** preferably utilizes WebObjects, and the database server **30** preferably utilizes Oracle. The three illustrated components of this server-side of the network **20** can be physically implemented in one or more hardware configurations and can reside together or at separate locations. They preferably communicate with each other through LAN or WAN connections depending on the physical location of each component.

[0060] The web server **26** manages and receives incoming requests from the local terminals **22** or local servers **24** and routes those requests to the appropriate server. The web server **26** is also operative to retrieve web pages of ASP's (application service providers) **96** (**FIG. 6**) or other URL's (uniform resource locators) **98** (**FIG. 6**) as appropriate for received requests.

[0061] Requests for business functions, such as statistical analysis, revenue calculations, and database queries for user data, are routed to the SOAP server, which runs in conjunction with the web server **26**. The SOAP server manages access to the business **28** and database **30** servers. The business server **28** is operable to perform revenue calculations, statistical analysis, and other GUI operations, described more fully below, and provides response messages and objects to other network components. To enable access to business information and modification of business rules, the business server **28** is preferably provided with an input device such as a key board and output devices such as a monitor **97** and printer **99**. In performing its operations, the business server **28** accesses the database server **30**, which stores user information and media information. The DBMS (database management system) operating the database server **30** preferably supports multimedia BLOB's (binary large object). The client-side local server **24**, when provided, provides the web server, SOAP server, business server, and database server functions at the local level, but the local server **24** is specifically configured for local use and cooperative processing with the central servers **26, 28, 30** and the local terminals **22**.

[0062] As illustrated in **FIG. 6**, the specific nature of the connections of the computer network **20A** between the server-side servers, collectively **100**, and numerous local installations **102, 104, 106, 108** of terminals **22** can be and is accomplished through many know communications means and is adaptable for connection with future improvements in telecommunications. The first local installation **102** utilizes a DSL (digital subscriber line) router **110** in operative communication with an ISP **112** (Internet service provider) or telephone company having DSL installations. It should be understood that other forms of broad bandwidth communications such as ADSL, VDSL, ISDN, and others can be used. The second local installation **104** utilizes a satellite **114** and router **116**. The third installation **106** also uses a DSL router **118**, but the router **118** is in operative communication with an infrastructure provider **120**. Further, the third installation **106** is provided with a local server **24** and a switching hub **122** for joining various components of the LAN. The fourth installation **108** utilizes a wireless provider **124**, a wireless receiver **126**, and router **128** to provide a mobile connection especially useful for marketing demonstrations and special events at temporary locations. Each of the installations **102-108** utilizes a wireless base station **130** for communication with the terminals **22**.

[0063] In the operation of the above described computer network **20A** illustrated in **FIGS. 7 and 8**, a unique local configuration file **140** containing context information for each terminal is retrieved and formatted at step **160** into an

XML request packet or message **142** and transmitted **162** over a computer readable transmission medium, as described in connection with **FIG. 6**, to the web server **26**. The context information includes, sponsor information, the location and owner of the terminal, and, for example, special event information for the location. The local configuration file **140** also contains an initial overall configuration for the GUI **32**. The web server **26** performs system login **164** at system startup. The XML request and return packets for system login contain the content information for the terminal logging in. The user **43** is then validated, also illustrated in step **164**. As users login to the system, the user validation step is repeated for each user with XML request and return packets containing content and user information. User validation also includes retrieval of information from the database server **30**. The database server **30** returns user data to the business server **28**, which generates, based on operator input rules and programming, the overall user interface at step **166**. The business server **28** then formats, at step **168**, the overall user interface into a return XML packet or message **144** and routes the return packet **144** to the web server **26** for transmission **170** to the requesting client application **45**. The client application **45** then reads the return XML packet **144** and implements the instructions therein for the application display process **148**.

[**0064**] The return XML packet **144** can also contain instructions for updating the local configuration file **140**. If update instructions are included, the client application **45** updates, for example, the user information and initial overall interface stored in the local configuration file **140** at step **172**. The return XML packet may also contain instructions for updating local media **63** and data **57** files. Thus, the computer network **20** provides a centrally programmed GUI **32**, so that when changes are desired, they are programmed at the central business server **28** level and implemented across the entire network **20** as various terminals **22** are activated.

[**0065**] To complete the user interface **32** (**FIG. 2**), the client application **45** sends out FTP (file transfer protocol) commands or file requests **146** for local and remote media files **63** and data files **57** and HTTP (hypertext transfer protocol) requests for network resources **65**. These commands **146** are sent in response to user action or in response to instructions in the return XML packet **144** to obtain these resources. As the client application **45** retrieves resources such as web pages and media players at step **174**, it begins the display process **148**. The display process **148** then continues throughout the user session to adapt the display interface **32** as the user **43** requests different applications.

[**0066**] Referring additionally to **FIGS. 9 and 10**, when, at step **176**, a user activates the interface control elements **190** to input a request for a different application or new content for a current application, the client application **45** optionally, at step **178**, formats and transmits a request to the pertinent application. The request is embedded in a URL, which triggers a request that is then routed to the network web server **26**. The business server **28** then develops, if required by the programmed business rules, a modified or updated overall user interface at step **180**. The return XML packet is encoded and transmitted **182** back to the requesting client application, which reads the return XML packet and modifies **184** or updates the user interface, local media files, and local data files as instructed.

[**0067**] Whenever the user **43** activates one of the interface control elements **190**, the control element issues a URL or application action request **192**, which is read by a container object command center **194**. The container object command center **194** is part of the client application software **45** and is operative to control container objects **196**, which correspond to screen application regions described below. If, for example, the user has requested a browser application, the command center determines at decision step **198** if a browser container object exists among the multiple container objects already in the system. If the browser container object already exits, the command center **194** targets the existing container object **196**, and it is used to display the browser application. The URL request **200** retrieves the identified web page for the browser. If the original application was something other than a browser, a file request may gather the local or network HTML resources **202** for display.

[**0068**] If there is not a browser container object among the exiting container objects, a container object builder **204**, which is itself an object residing on the client application software **45**, reads the appropriate XML layout file **206** from the database server **30**. The client application software **45** then creates the new browser container object **208**. An environment controller **210** controls the layered media environment and gathers the local and/or network media files **202** to populate the requested browser container object. Once operating, the browser container object is capable of interpreting, rendering, and allowing user interaction with HTML and XML resources. Thus, the computer network **20** provides a user interface **32**, which changes and updates on the fly in response to user input.

[**0069**] Referring to **FIGS. 2, 3, 7**, and **10**, the overall user interface **32** includes at least one but preferably a plurality of screen application regions corresponding to the container objects **208** in the software. Preferably, the screen application regions include a control region **220**, a pair of advertiser link regions **222, 224**, a features and favorites links region **226**, an audio player region **228**, and a primary application region **230**. The screen applications regions **220-230** are generally rectangular. A status area **232** provides information and event notices, such as new mail and/or buddy logged on, to the user. In the embodiment of the screen shown in **FIG. 2**, the interface also includes an application status region **236**, displaying a throbber, which indicates the status of the application. The screen application regions **220-232** are defined by the client application **45** pursuant to instruction received from the network business server **28**. Specifically, the client application **45** creates a global grid consisting of the one or more screen application regions. Each screen application region is defined by the configuration instructions to contain properties and functions applicable to the media type to be displayed within it. The client application software is able to support and the business server contains XML layout files for QuickTime, HTML, Shockwave, and other media formats. However, this capability can be extended as new media types are developed and built into XML configuration.

[**0070**] Referring again to **FIG. 2**, the client application **45** (**FIG. 3**) places one or more navigation control **240** in the control region **220**. The navigation control **240** comprises an interactive movie, preferably an Apple, QuickTime movie, having control elements, graphic elements, and embedded commands. The control elements include application tabs

242 and control buttons 244 with the screen regions pertaining to the control elements being identified for activation by a pointer. There are preferably sixteen application tabs including home, game, movie, music, e-mail, Internet, chat, and search tabs among others. The control buttons 244 illustrated are browser control buttons and include forward, back, stop, reload, scroll up, and scroll down functions. Several other control button configurations are also provided. For example, when a movie media is active in the primary application region 230, the control buttons are changed to play, stop, rewind/reverse, fast forward, slow forward, and pause. As a user activating the various control tabs 242 selects different applications, the control buttons 244 change. The change is instantly accomplished by playing a different interactive movie. To provide these dynamic button controls 244 in a coded fashion would require recompiling code, which is far slower than changing a 50 k movie. The invention also contemplates playing multiple interactive movies simultaneously, and interactive movies that change state to suite the new context without loading a new movie.

[0071] The pair of advertiser link regions 222, 224, the features and favorites links region 226, and the audio player region 228 are descriptively named and include configurations instructions to support these applications. However, any type of application could be provided in these screen application regions. When, for example, a user "clicks" on one of the advertiser links, the browser is launched, as described above, and the browser is displayed in the primary application region 230. The control buttons 244 for the browser are also displayed. The user 43 can select a control button 244 to issue commands to the primary application object. For example, the user 43 may choose to stop the load or playback of a file by clicking on the stop control button, while activating the reload control button will refresh the URL last requested. The media files 63 associated with the control elements 242, 244 include the media (graphics, images, and sounds), the control elements, and instructions embedded in the movie and triggered by user activation in the user interface or an event such as a timer.

[0072] As stated above, the environment is a layered media environment. To that end a background movie 248 is played beneath substantially all of the screen application regions. The background movie provides the borders and partitions of the various screen application regions. The other interactive applications are then overlaid onto the background movie, which is preferably a static image without direct interactive controls. A browser application 250 (FIG. 11H) is then operated in a selected one of the plurality of screen application regions. Preferably, the browser is operated in the primary screen application region 230, and is provided with a top browser tab 246. Multiple applications, preferably up to five, can be operated in the primary screen application region. Each application has its own identification tab, so that a user can easily bring the desired application to the front of the primary application region 230. The identification tabs are preferably provided by an interactive movie.

[0073] With the background movie playing behind the primary application area and other interactive control and/or display elements, the browser functions as an integrated application within the media environment. Embedding the browser gives the user a consistent user interface and on-screen representation of the environment (other images and controls) while ensuring that files displayed within the browser retain their native format and functionality. This method also allows the browser to be controlled by the media environment, either by application events, events contained in other movie areas or user action on interactive control elements within other movie components. The application gives the browser object both the appearance and function of an embedded applet which has full native capability as well as interactive capability with the rest of the environment and other components contained therein.

[0074] To further illustrate the present invention, FIG. 11A through FIG. 11J illustrate an exemplary user session, which will be briefly described. In FIG. 11A the user 43 enters his or her name and password and swipes either a prepaid or credit card in a provided magnetic card reader (not shown). Upon validation by the network servers 26, 28, 30, the user is greeted by the initial screen of FIG. 11B. This is the entry page for the environment. FIG. 11C illustrates the user moving the pointer 251 over the music tab 252 which highlights the music tab and displays text about the action associated with the tab in the main window text area 254. The specific text displayed in this example is "listen to music for many formats." After clicking the pointer device button, the user goes to the music homepage 258 illustrated in FIG. 11D. This is an internal page with links to other media within the music category. At this point, the browser control buttons 244 are still displayed in the interactive control movie. In FIG. 11E the user selects the music video button for "smooth" at location 256. A movie application 260 illustrated in FIG. 11F is then presented in the primary application region over the main music page and in its own panel. The interactive control movie 262 for a movie application is loaded and the movie media is loaded and starts playing "smooth." In FIG. 11G the user then highlights the Internet tab 264 to go to the World Wide Web. Upon clicking the button, a new web browser page is created and displayed in a new panel with the web browser therein. In the illustration displayed in FIG. 11H the user has selected the coffee universe though any URL could be selected. In FIG. 11I the user selects the Saturn ad banner 266 in the top advertising region and clicks. A forth web panel 268 is opened as illustrated in FIG. 11J, and the user is taken to the selected URL for Saturn.

[0075] Another user session in a commercial setting is illustrated in FIG. 12A through FIG. 12I. For this session it is understood that the user is in the context of a retail establishment. Thus the local configuration file provides a completely different starting point for this retail venue. FIG. 12A illustrates the start of the user session with a motion picture commercial 270. The user interacts with the system by pressing any key or clicking the provided pointer device and the introduction screen illustrated in FIG. 12B replaces the movie. The introduction screen is an interactive display designed to guide the user to other areas of the system. In FIG. 12C the user is illustrated selecting the t-shirt option 272 from the screen thereby changing the graphics on the main page to the selected item as they point to each of the items shown. In FIG. 12D the user has selected the sweater section 274 and sees a preview of the selection. These views are all Flash based. Upon selecting the sweater section a talking head attendant movie 276 illustrated in FIG. 12E appears and begins introducing the selection. The talking head is provided in the form of a movie. In FIG. 12F the attendant movie clip 276 has automatically taken the user to

more information as the narrative moves along without user interaction. Thus the movie is controlling the other application piece, that is, the display of the item for sale. In **FIG. 12G** the graphic content is replaced with another movie **278** which is synchronized to the actions of the attendant movie. **FIG. 12H** illustrates a VR (virtual reality) movie **280**, which has replaced the synchronized movie from **FIG. 12G**. The user can move around and interact with the VR movie. When an item of interest is finally selected, the VR movie is replaced by a web browser component **282** illustrated in **FIG. 12I**. Again the attendant movie explains and interacts with the browser.

[0076] The above described computer network **20** and GUI **32** are preferably implemented through either a client purchase agreement business model **300** (**FIG. 13**) or a client subscription agreement business model **302** (**FIG. 14**). In **FIG. 13**, the client purchase agreement includes four (4) sources of revenue: hardware **304**, service **306**, utilization **308**, and advertising **310**. The network operator **312** receives hardware revenue **304** from the client. Specifically, the client is the owner or operator of the sports bar or other establishment utilizing the computer network **20** and having terminals **22** installed at the bar. The hardware revenue **304** is in the form of equipment purchase or lease, and as the computer network **20** is establish and grows, the hardware revenue also derives from hardware upgrades. The hardware revenue **304** also includes revenues paid by the client for installation of the hardware. The service contract revenue **306** is simply income from required service on the hardware. The services generating the service revenue **306** can be provided directly by the network operator or these services are contracted out to a local service company. The regular recurring utilization revenue derives from customers using the system. Customers pay for system/terminal use with a credit card, prepaid card, or a monthly-billed account, which bills on a use basis or a flat fee. The regular recurring advertising revenue **310** is received from advertisers and is based on a display of ads in one of screen advertiser link regions **222**, **224**. Additionally or alternatively, the advertising revenue **310** can also be based on customer purchases.

[0077] Preferably on a monthly basis, a monthly account statement **314** is generated. The monthly account balance credits, at step **316**, the client for the recurring utilization and advertising revenue. A credit **318**, preferably electronic, is given to the client, and an electronic monthly statement **320** is transmitted.

[0078] The subscription agreement business model **302** illustrated in **FIG. 14** operates similarly to the purchase model **300**. The recurring utilization **322** and advertising **324** revenues are received as described, but the only other revenue received is the subscription down payment **326**. At system startup, the client pays the first month subscription fee or down payment **326** to the network operator **328**. In the monthly account balance process **330**, the client again receives a credit **332** for the recurring utilization and advertising revenues **322**, **324**, but there is also a debit **334** for the monthly subscription fee. If the revenue credit **332** is greater than the subscription fee debit **334**, then the client account receives an electronic credit for the difference at step **336**. If however, the debit **334** is greater than the revenue credit **332**, the client account receives an electronic debit for the difference at step **336**. The computer system **20** then generates a monthly account statement **338** reflecting the credit or debit to the client account.

[0079] When a multimedia event occurs at one location and it is broadcast to other locations, revenue is distributed as illustrated in **FIG. 15**. Utilization **340** and advertiser **342** revenues are paid to the network operator **344**. Alternatively a client fee **346** is also paid to the network operator **344**. The network operator **344** then distributes the revenue to the band and/or promoter **348** and the broadcaster **350**. The network operator **344** also keeps a share of the income. The client **352**, alternatively also receives a portion of the income or if the client's share **352** of the income is greater than the client fee **346**, the client receives some of the income.

[0080] Thus, a context sensitive, dynamic user interface is disclosed which is programmed and developed from a central location and utilizes interactive control movies (instead of hard programming), browsers embedded within movies, and application layering in single screen application regions to provide a versatile computer network thereby increasing expansion of revenue generating computers into nontraditional venues. While preferred embodiments and particular applications of this invention have been shown and described, it is apparent to those skilled in the art that many other modifications and applications of this invention are possible without departing from the inventive concepts herein. It is, therefore, to be understood that, within the scope of the appended claims, this invention may be practiced otherwise than as specifically described, and the invention is not to be restricted except in the spirit of the appended claims. Though some of the features of the invention may be claimed in dependency, each feature has merit if used independently.

[0081] Glossary

[0082] Applet—a program designed to be executed from within another application. Unlike applications applets cannot be executed directly from the operating system.

[0083] Application—a program or group of programs designed for end users. Applications include programs such as browsers, word processors, and spreadsheets.

[0084] ASP—application service provider—third party entities that manage and distribute software based services and solutions to customers across a WAN from a central data center.

[0085] Bandwidth—the amount of data that can be transmitted in a fixed amount of time. For digital devices the bandwidth is typically expressed in bits per second (bps) and, for analog devices the bandwidth is typically expressed in cycles per second (Hz).

[0086] BLOB—binary large object—a collection of binary data stored as a single entity in a database management system. BLOB's are used primarily to hold multimedia objects such as images, videos, and sound, though they can also be used to store programs.

[0087] Cache—a special high-speed storage mechanism. Frequently accessed files are stored in memory for quick access by an operating application.

[0088] Container Object—is an object that acts as a holding container for one or more other objects (movies, browsers, etc.).

[0089] CPU—central processing unit—the CPU is the brains of the computer and is where most calculations take place. Typically the CPU is housed in a single-chip microprocessor.

[0090] CRT—cathode ray tube—the technology used in most televisions and computer display screens. A CRT works by moving an electron beam back and forth across the back of the screen to illuminate phosphor dots on the inside of the glass tube.

[0091] DBMS—database management system—a collection of programs that enables the storage, modification, and extraction of information from the database.

[0092] DSL—digital subscriber lines—there are many types of DSL technologies relating to high data rate transfer over existing copper telephone lines.

[0093] Embedded Object—an object created with one application and embedded into a document created by another application. Embedding the object in contrast to simply inserting or pasting it in, ensures that the object retains its original format. The embedded object can be modified with the original program.

[0094] FTM—flat technology monitor—while conventional display screens are curved, flat technology monitors have a flat display screen to reduce glare.

[0095] FTP—file transfer protocol—guidelines used on the Internet for sending files.

[0096] GUI—graphical user interface—a program interface that takes advantage of the computer's graphics capabilities to make the program easier to use by presenting controls and options to the user. GUI's include several basic components including a pointer, which is a symbol such as a small, angled arrow, which appears on the display screen and it is moved about by a pointing device such as a mouse or a trackball.

[0097] HTTP—hypertext transfer protocol—guidelines defining how messages are formatted and transmitted over the Internet.

[0098] Hub—a common connection for devices in a network. Hubs are commonly used to connect segments of a LAN and typically include multiple ports.

[0099] IDE—intelligent drive electronics or integrated drive electronics—an IDE is an interface for storage devices in which the controller is typically integrated into the disk or CD Rom drive.

[0100] ISDN—integrated services digital network—an international communications standard for sending voice, video, data over digital telephone lines or normal telephone wires. ISDN supports increased data transfer rates with modern versions of ISDN supporting transmission rates of up to 1.5 mbps (mega bytes per second).

[0101] LAN—local area network—a computer network that spans a relatively small area typically in a single building or group of buildings.

[0102] LCD—liquid crystal display—a type of display used in many portable computers. LCD displays utilize two sheets of polarizing material and has liquid crystal solution between them. An electric current passed through the liquid causes the crystals to align so that light cannot pass through them. Thus, each crystal operates like a shutter.

[0103] Movie—a media file viewed on screen in a player type environment and having controls that are operable to start, stop, and perform other variables of the file. The movies according to the present invention may have either static or motion images and may contain interactive elements or controls.

[0104] Object—a broad term including any item than can be individually selected and manipulated. In the context of this application it is referred to more specifically as a self-contained entity that consists of both data and procedures to manipulate the data.

[0105] OS—operating system—the most important program that runs on a computer. Operating systems perform basic tasks such as recognizing input from the keyboard and sending output to the display screen and printers. They are also responsible for controlling peripheral devices such as disk drives.

[0106] Router—a device that connects any number of LAN's and uses headers and a forwarding table to determine where packets and messages go and what is the best route to be taken.

[0107] SCSI—small computer system interface—SCSI is a parallel interface standard for attaching peripheral devices such as hard drives to computers.

[0108] SOAP—simple object access protocol—SOAP provides a way for applications to communicate with each other over the Internet independent of the specific platform of the computer systems.

[0109] SVGA—super video graphics array—a graphics display system for PC's providing high resolution and a color palette of up to 16 million colors. Typical SVGA monitors provide sufficient memory to display 256 colors simultaneously.

[0110] URL—uniform resource locator—the global address of documents and other resources on the World Wide Web.

[0111] VPN—virtual private network—a network that is constructed by using public wires to connect nodes. These systems use encryption and other security mechanisms to assure that only authorized users can access the network and that the data cannot be intercepted.

[0112] WAN—wide area network—a computer network that spans a relatively large geographic area. Typically a WAN consists of two or more LAN's and may be connected through the public switch telephone network.

What is claimed is:

1. A dynamic user interface display comprising:

a plurality of screen application regions; and

an interactive movie, including control elements, and being displayed in a selected one of the plurality of screen application regions.

2. The interface according to claim 1 further comprising a background movie played beneath substantially all of the

screen application regions, and a browser operating in a selected one of the plurality of screen application regions and overlaying the background movie.

3. The interface according to claim 1 wherein the control elements comprise application tabs and control buttons.

4. The interface according to claim 3 further comprising a plurality of control button sets and wherein the control buttons comprise dynamic control buttons operative to change between sets based on a selected application.

5. The interface according to claim 1 further comprising an advertisement displayed in a second selected one of the plurality of screen application regions.

6. The interface according to claim 5 wherein the advertisement comprises a link to a web site corresponding to the advertising.

7. The interface according to claim 3 wherein the application tabs include an e-mail application tab, game application tab, movie application tab, search application tab, music application tab, and home application tab.

8. The interface according to claim 3 wherein the control buttons include a play control button, pause control button, reverse control button, and fast forward control button.

9. The interface according to claim 3 wherein the control buttons include a back control button, reload control button, forward control button, a stop control button, a scroll down control button, and a scroll up control button.

10. A dynamic user interface display comprising:

a plurality of screen application regions;

a background movie played beneath substantially all of the screen application regions;

a browser operating in a selected one of the plurality of screen application regions and overlaying the background movie.

11. The interface display according to claim 10 further comprising an interactive movie, including control elements, and being displayed in a selected one of the plurality of screen application regions.

12. The interface display according to claim 10 wherein the browser includes a browser identification tab.

13. The interface display according to claim 12 further comprising an application also operating in the selected one of the plurality of screen application regions and the application having an application identification tab.

14. The interface display according to claim 10 further comprising up to four additional applications operating in the selected one of the plurality of screen application regions.

15. A method for controlling a dynamic user interface, the method comprising:

playing a control movie having graphical elements and interactive control elements for controlling an application;

selecting a control element based on user input; and

managing the application as directed by the control element.

16. The method according to claim 15 further comprising changing the interactive control elements to a second set of interactive control elements for controlling a different application.

17. The method according to claim 15 wherein the movie comprises a browser control movie, and further comprising changing to a video control movie upon selection of a video application by a user.

18. The method according to claim 15 further comprising providing a plurality of screen application regions, playing the control movie in a selected one of the plurality of screen application regions, and displaying other applications in the remaining screen application regions, the other applications including HTML applications, Shockwave applications, and QuickTime applications.

19. The method according to claim 18 wherein the control movie comprises a QuickTime movie.

20. A method for providing a dynamic user interface, the method comprising

defining a plurality of screen application regions;

playing a background movie beneath substantially all of the screen application regions; and

operating a browser in a selected one of the plurality of screen application regions.

21. The method according to claim 20 further comprising playing a control movie, including control elements, in a second one of the selected control regions.

22. The method according to claim 21 further comprising controlling the browser with the control elements of the control movie.

23. The method according to claim 20 further comprising displaying an advertisement in a second one of the selected control regions.

24. The method according to claim 20 further comprising operating multiple applications in the selected one of the plurality of screen application regions, and displaying tabs for each of the applications and the browser.

25. A method for providing a dynamic user interface, the method comprising:

defining a plurality of screen application regions; and

operating multiple applications in a selected one of the plurality of screen application regions, and displaying tabs for each of the applications.

26. The method according to claim 25 wherein at least one of the applications is a browser.

27. The method according to claim 25 further comprising playing a background movie beneath substantially all of the screen application regions.

28. The method according to claim 25 further comprising playing a control movie having graphical elements and interactive control elements for controlling one of the applications.

29. The method according to claim 28 further comprising changing to another control movie having graphical elements and interactive control elements for controlling another one of the applications.

30. A method for generating a dynamic user interface, the method comprising:

transmitting local configuration information, including context information, from a local terminal to a server;

determining, based on the context information, an overall user interface configuration;

determining, based on the context information, applications for display in the overall user interface configuration; and

transmitting the overall user interface configuration and applications for display in the overall user interface configuration to the local terminal.

31. The method according to claim 30 further comprising formatting data from the local configuration information into a request XML packet for transmission from the local terminal to the server, and formatting the overall user interface configuration and the applications for display in the overall user interface configuration into a return XML packet for transmission to the local terminal.

32. The method according to claim 30 further comprising retrieving local resources for display in the overall user interface configuration.

33. The method according to claim 30 wherein determining the overall user interface configuration comprises using a locally stored overall user interface configuration.

34. The method according to claim 30 further comprising retrieving Internet resources for display in the overall user interface configuration.

35. The method according to claim 30 further comprising updating the local configuration file.

36. The method according to claim 30 wherein the overall user interface configuration includes a plurality of screen application regions, and further comprising opening, in response to a user request, an application in a selected one of the plurality of screen application regions, generating a control movie operative to control the application, and playing the control movie in another selected one of the plurality of screen application regions.

37. The method according to claim 36 further comprising opening, in response to another user request, another application, generating another control movie operative to control the another application, and playing the another control movie.

38. The method according to claim 36 further comprising opening, in response to another user request, another application in the selected one of the plurality of screen application regions, generating another control movie operative to control the another application, and playing the another control movie in the another selected one of the plurality of screen application regions.

39. The method according to claim 38 further comprising displaying a tab for the application and displaying another tab for the another application.

40. A computer readable medium containing instructions for controlling a computer network to display a dynamic media interface, comprising:

defining a plurality of screen application regions on a computer display;

playing a background movie beneath substantially all of the screen application regions; and

operating a browser in a selected one of the plurality of screen application regions.

41. A computer readable medium containing instructions for controlling a computer network to display a dynamic media interface, comprising:

playing a control movie having graphical elements and interactive control elements for controlling an application;

selecting a control element based on user input; and

managing the application as directed by the control element.

42. A computer readable medium containing instructions for controlling a computer network to display a dynamic media interface, comprising:

defining a plurality of screen application regions; and

operating multiple applications in a selected one of the plurality of screen application regions, and displaying tabs for each of the applications.

43. A method in a computer system for displaying a dynamic media interface, the method comprising:

defining a plurality of screen application regions on a computer display;

playing a background movie beneath substantially all of the screen application regions;

operating a browser in a selected one of the plurality of screen application regions;

operating multiple applications in the selected one of the plurality of screen application regions;

displaying tabs for each of the applications and the browser; and

selectively playing a control movies having graphical elements and interactive control elements for controlling the browser and the applications.

44. The method according to claim 43 wherein displaying tabs for each of the applications and the browser comprises displaying top tabs for each of the applications and the browser.

45. A computer installation comprising:

a substantially rigid wall frame member;

a substantially constant wall cover member attached to the wall frame member and including an outer side and an inner side facing the wall frame member;

a computer housing mounted on the wall frame member adjacent the inner side of the wall cover member;

a central processing unit supported by the computer housing;

a computer display in operative communication with the central processing unit and being controlled by the central processing unit, the computer display being positioned adjacent the outer side of the wall cover member; and

an input device accessible by a user and the input device being in operative communication with the central processing unit.

46. The computer installation according to claim 45 wherein the central processing unit comprises a modular central processing unit removable from the housing for replacement by another modular central processing unit.

47. The computer installation according to claim 45 wherein the computer display is mounted on the outer side of the wall cover member.

48. The computer installation according to claim 45 wherein the computer display is mounted on the wall frame member.

49. The computer installation according to claim 45 wherein the computer display is mounted on the computer housing.

50. The computer installation according to claim 45 further comprising a second wall frame member and wherein the computer housing is mounted to both wall frame members.

51. The computer installation according to claim 45 wherein the input device is in remote operative communication with the central processing unit.

52. The computer installation according to claim 45 further comprising at least one speaker in operative communication with the central processing unit and the at least one speaker being positioned adjacent the outer side of the wall cover member.

53. The computer installation according to claim 45 further comprising a central server in operative communication with the central processing unit, and wherein the central processing unit comprises a local central processing unit.

54. The computer installation according to claim 53 further comprising a local server in operative communication between the central server and the local central processing unit, and the local server is in operative communication with the central server through the Internet.

55. A computer network comprising

a central server;

a plurality of local terminals in operative communication with the central server, and the local terminals including local displays and local input devices;

the local displays being operative to display a plurality of screen application regions, a background movie beneath substantially all of the screen application regions, and a browser in a selected one of the plurality of screen application regions.

56. The computer network according to claim 55 further comprising a local server in operative communication between the central server and the local terminals.

57. The computer network according to claim 55 wherein the input devices comprise keyboards having illuminated keys.

58. The computer network according to claim 55 wherein the local terminals further include pointing devices.

59. The computer network according to claim 58 wherein the pointing devices comprise relative pointing devices.

60. The computer network according to claim 59 wherein the relative pointing devices comprise track balls.

61. A computer network comprising:

a central server;

a plurality of local terminals in operative communication with the central server, and the local terminals including local displays and local input devices;

the local displays being operative to display a plurality of screen application regions and display in a selected one of the plurality of screen application regions a control movie having graphical elements and interactive control elements for controlling an application.

62. The computer network according to claim 61 wherein the displays are further operative to display a plurality of control movies in the selected one of the plurality of screen application regions, each of the control movies having interactive control elements for controlling different applications.

63. A computer network comprising

a central server;

a plurality of local terminals in operative communication with the central server, and the local terminals including local displays and local input devices;

the local displays being operative to display a plurality of screen application regions and display in a selected one of the plurality of screen application regions multiple applications including a browser.

64. The computer network according to claim 63 wherein the display is further operative to display tabs corresponding to the applications.

65. A computer network for driving applications using an interactive movie, the network comprising:

a central server;

a plurality of local terminals including local displays; and

an interactive movie including control elements displayed on the local displays.

66. The network according to claim 65 further comprising a plurality of interactive movies for selective display on the local displays.

67. The network according to claim 65 wherein the plurality of local terminals are located at a first location and further comprising another plurality of local terminals are located at a second location.

68. A computer readable data transmission medium containing data stricture comprising:

a first portion identifying context information including terminal location information;

a second portion identifying user information including user identification and user preferences.

69. The transmission medium according to claim 68 wherein the context information further includes special event information for the terminal location.

70. The transmission medium according to claim 68 wherein the context information and the user information are in XML format.

71. A computer readable data transmission medium containing data structure comprising:

a first portion identifying an overall user interface configuration;

a second portion identifying at least one application for display in the overall interface configuration.

72. The transmission medium according to claim 71 further comprising a third portion identifying local resources for display in the overall interface configuration.

73. The transmission medium according to claim 71 further comprising a third portion identifying Internet resources for display in the overall interface configuration.

74. The transmission medium according to claim 71 wherein overall user interface configuration and the at least one application for display in the overall interface configuration are in XML format.

75. A method in a computer network for communicating with a central server, the method comprising:

receiving context information, including terminal location, from a local terminal;

in response to receiving the context information, determining an overall interface configuration; and

receiving from the central server, the overall interface configuration.

76. The method according to claim 75 wherein the context information further includes a locally stored, initial overall interface configuration.

77. A method for distributing income from transmission of a media event to a venue open to customers, the method comprising:

a network operator receiving income from the customers for viewing the media event;

the network operator retaining a network operator portion of the income;

the network operator distributing a promoter portion of the income to a promoter;

the network operator distributing a venue portion of the income to an owner of the venue.

78. The method according to claim 77 further comprising the network operator distributing a broadcaster portion of the income to a broadcaster.

* * * * *

US 20020029261A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2002/0029261 A1

Shibata (43) Pub. Date: Mar. 7, 2002

(54) INFORMATION SUPPLY SYSTEM AND CONTROL METHOD THEREOF

(76) Inventor: Junji Shibata, Tokyo (JP)

Correspondence Address:
Ronald L. Grudziecki
BURNS, DOANE, SWECKER & MATHIS, L.L.P.
P.O. Box 1404
Alexandria, VA 22313-1404 (US)

(21) Appl. No.: 09/946,590

(22) Filed: Sep. 6, 2001

(30) Foreign Application Priority Data

Sep. 7, 2000 (JP) 2000-272143

Publication Classification

(51) Int. Cl.[7] G06F 15/16

(52) U.S. Cl. 709/219; 709/218

(57) ABSTRACT

To provide an information supply system that supplies desired information to service subscribers, capable of automatically selecting and supplying information required by the user from whole information. Local servers 4 to 6 are provided with user DB 8 to 10 that store reception settings set by individual service subscribers. When service subscriber terminal moves to a service area of an access point connected, a local server establishes a channel with the service subscriber terminal, and references the corresponding reception setting from a user DB and only presents the message that matches the reception setting from message data stored in the local server to the service subscriber terminal.

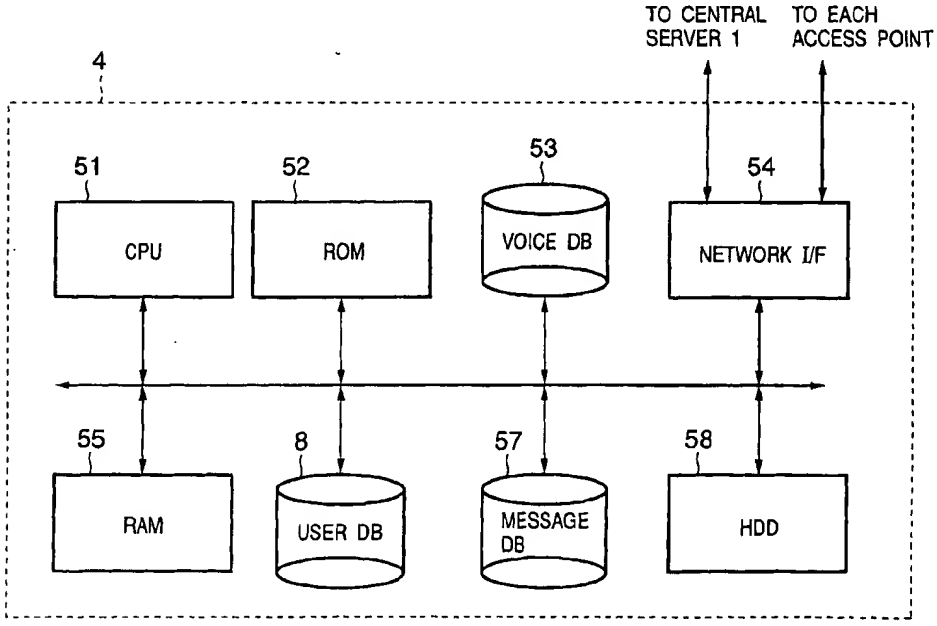


FIG. 1

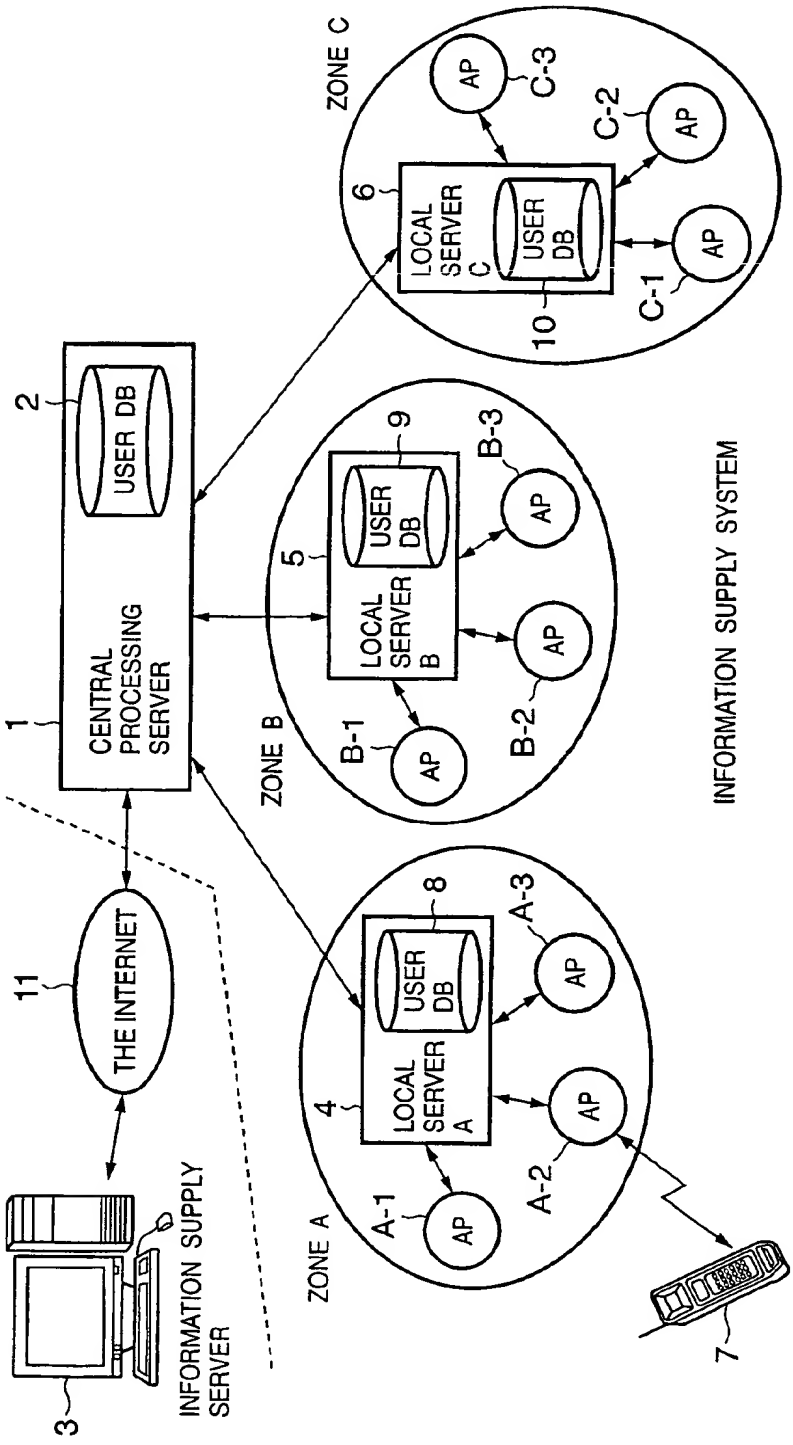

INFORMATION SUPPLY SYSTEM
CENTRAL PROCESSING SERVER
USER DB
LOCAL SERVER A
LOCAL SERVER B
LOCAL SERVER C
ZONE A
ZONE B
ZONE C
AP
THE INTERNET
INFORMATION SUPPLY SERVER
1
2
3
4
5
6
7
8
9
10
11
A-1
A-2
A-3
B-1
B-2
B-3
C-1
C-2
C-3

FIG. 2

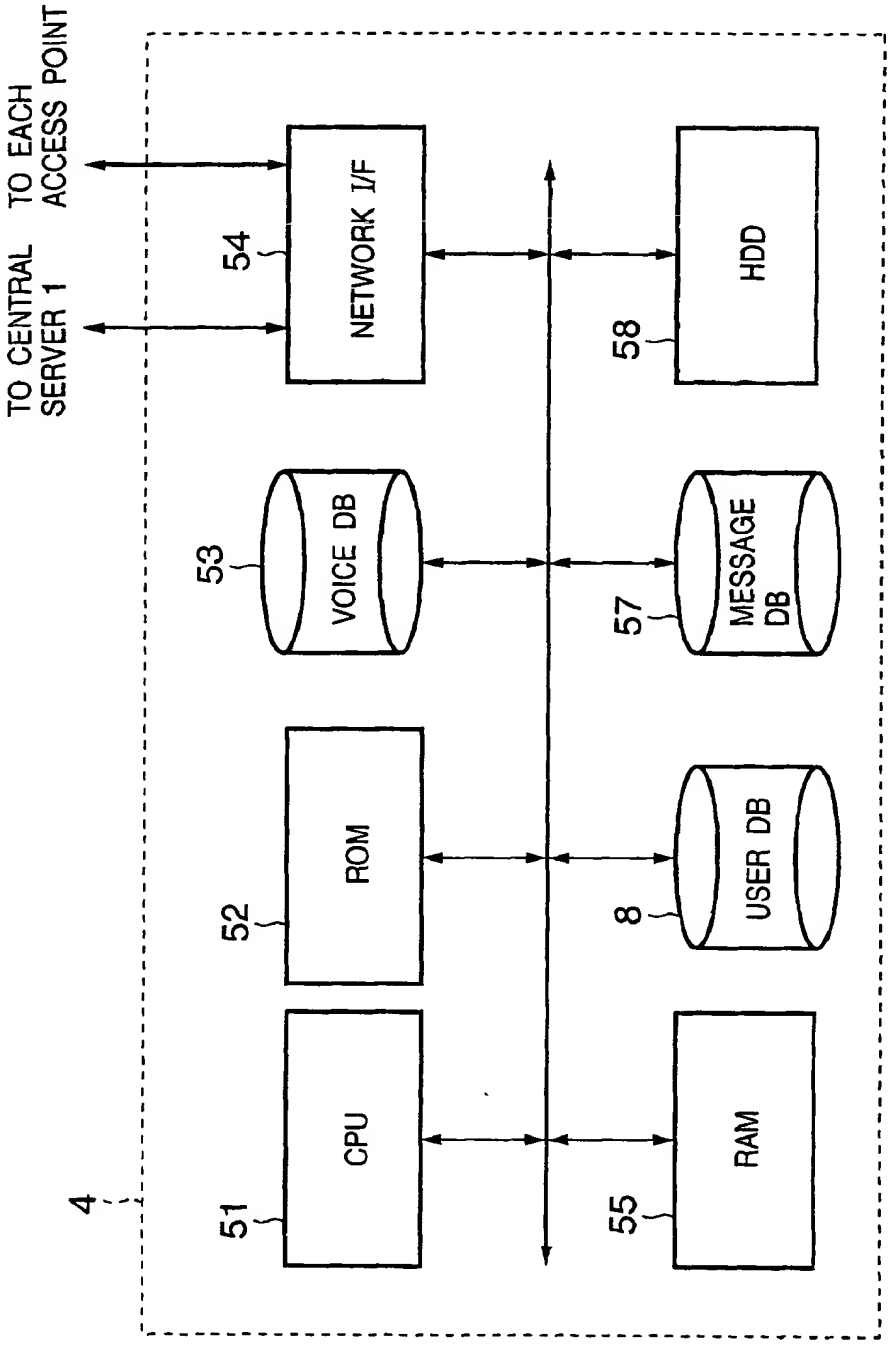

TO CENTRAL SERVER 1
TO EACH ACCESS POINT
NETWORK I/F
54
HDD
58
VOICE DB
53
MESSAGE DB
57
ROM
52
USER DB
8
CPU
51
RAM
55
4

FIG. 3

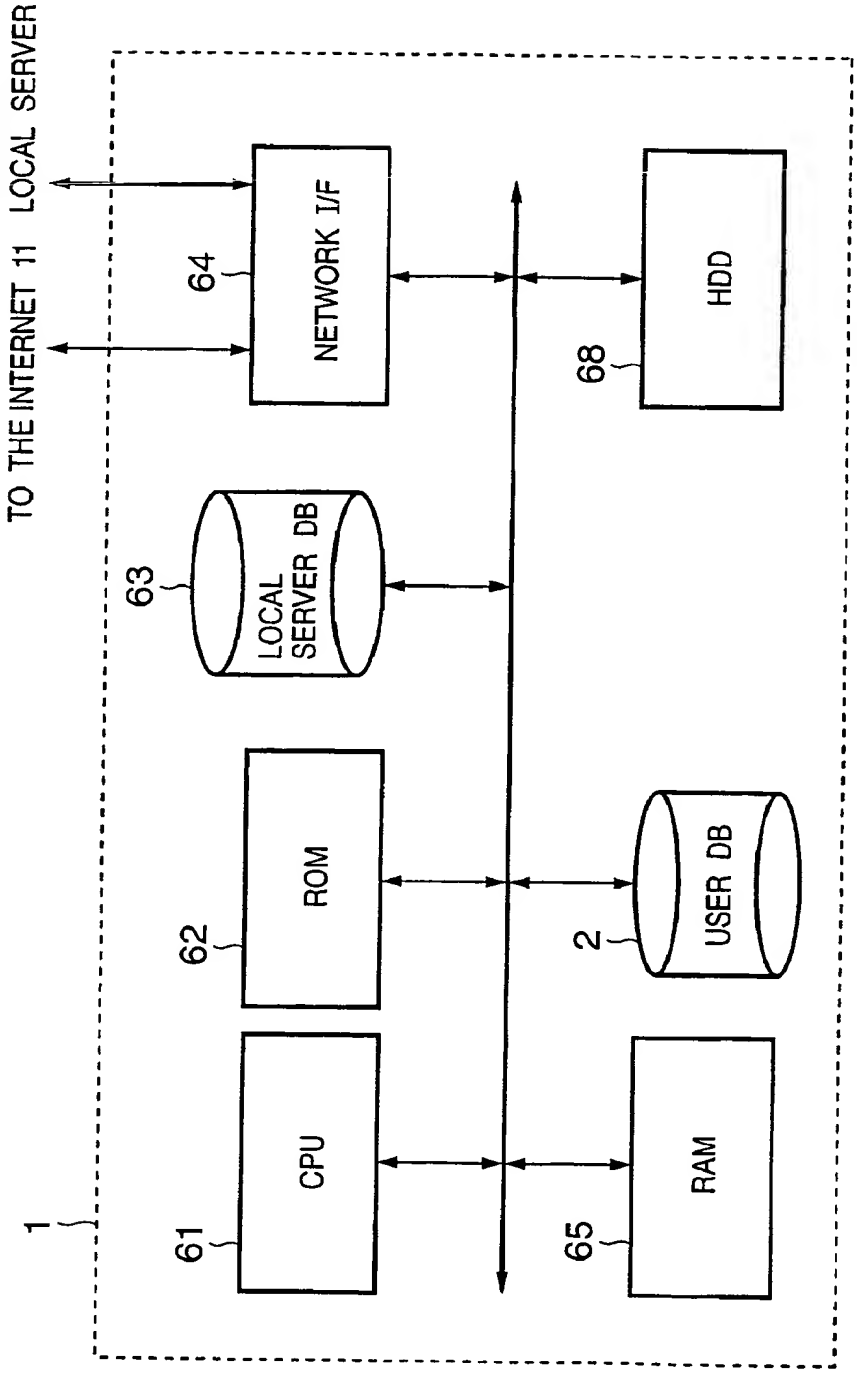

TO EACH LOCAL SERVER
TO THE INTERNET 11
64
NETWORK I/F
68
HDD
63
LOCAL SERVER DB
62
ROM
2
USER DB
61
CPU
65
RAM
1

FIG. 4

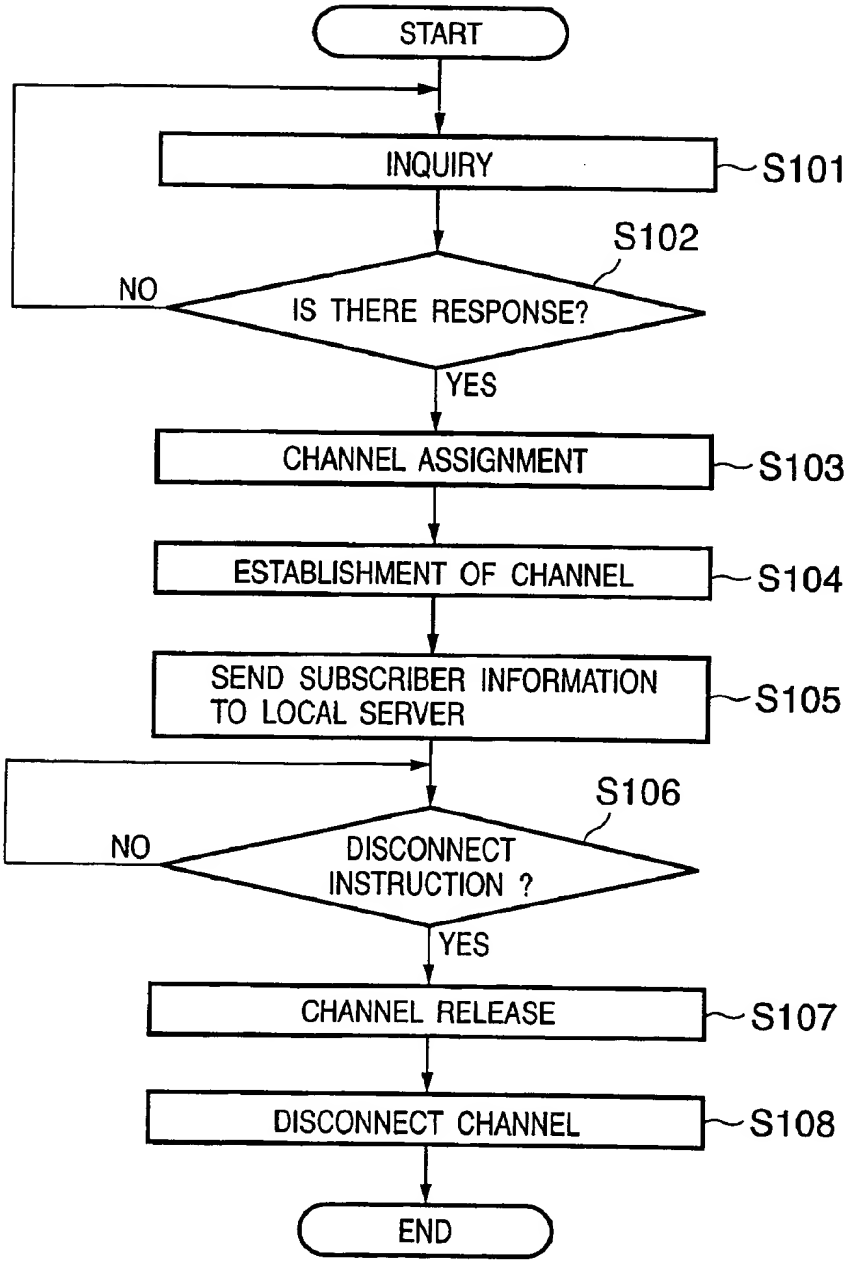
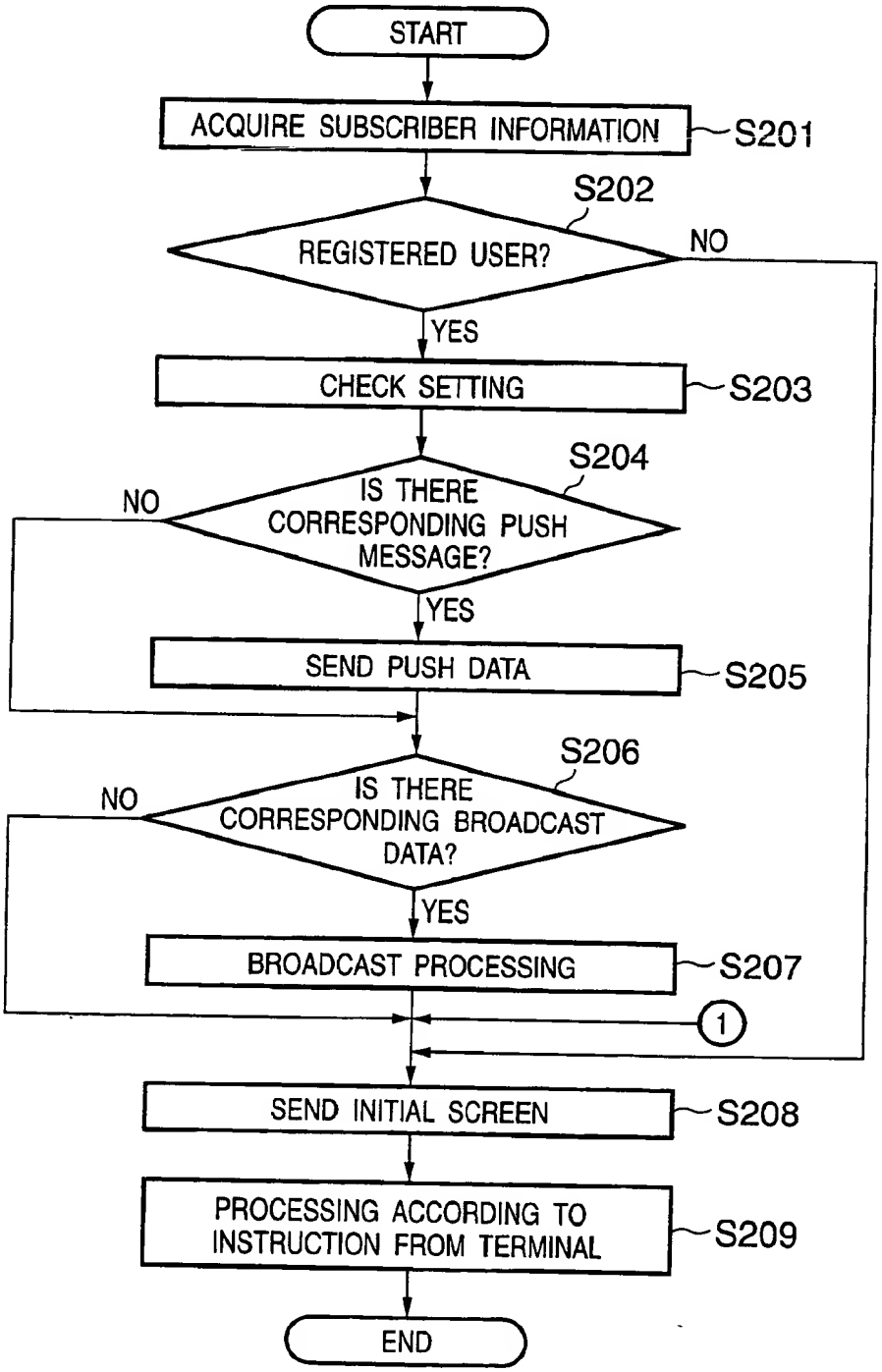

START
INQUIRY
S101
S102
NO
IS THERE RESPONSE?
YES
CHANNEL ASSIGNMENT
S103
ESTABLISHMENT OF CHANNEL
S104
SEND SUBSCRIBER INFORMATION TO LOCAL SERVER
S105
S106
NO
DISCONNECT INSTRUCTION ?
YES
CHANNEL RELEASE
S107
DISCONNECT CHANNEL
S108
END

FIG. 5


START
ACQUIRE SUBSCRIBER INFORMATION
S201
S202
REGISTERED USER?
NO
YES
CHECK SETTING
S203
S204
IS THERE CORRESPONDING PUSH MESSAGE?
NO
YES
SEND PUSH DATA
S205
S206
IS THERE CORRESPONDING BROADCAST DATA?
NO
YES
BROADCAST PROCESSING
S207
1
SEND INITIAL SCREEN
S208
PROCESSING ACCORDING TO INSTRUCTION FROM TERMINAL
S209
END

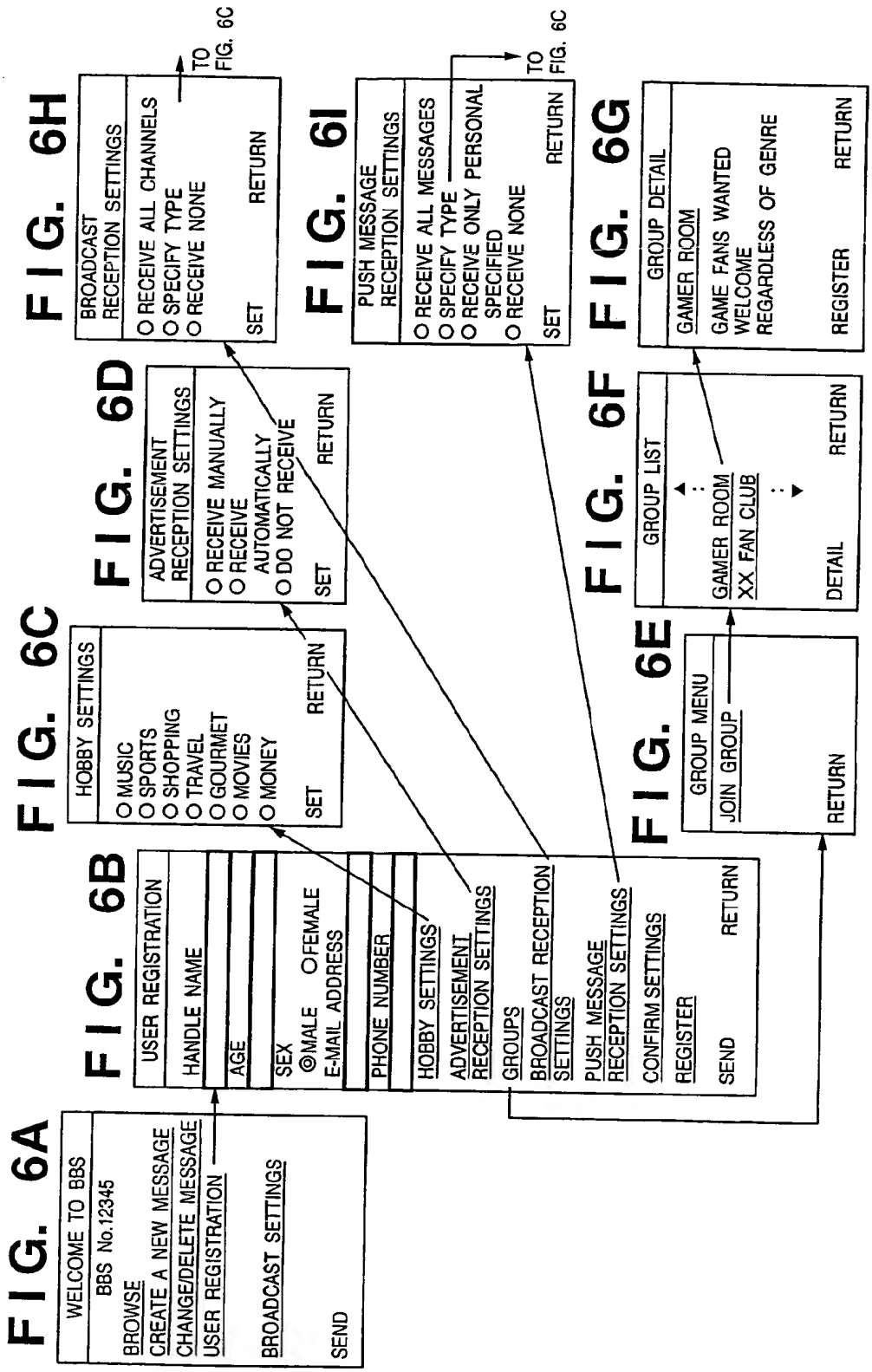


FIG. 6A
WELCOME TO BBS
BBS No.12345
BROWSE
CREATE A NEW MESSAGE
CHANGE/DELETE MESSAGE
USER REGISTRATION
BROADCAST SETTINGS
SEND
FIG. 6B
USER REGISTRATION
HANDLE NAME
AGE
SEX
◎MALE ○FEMALE
E-MAIL ADDRESS
PHONE NUMBER
HOBBY SETTINGS
ADVERTISEMENT RECEPTION SETTINGS
GROUPS
BROADCAST RECEPTION SETTINGS
PUSH MESSAGE RECEPTION SETTINGS
CONFIRM SETTINGS
REGISTER
SEND
RETURN
FIG. 6C
HOBBY SETTINGS
○ MUSIC
○ SPORTS
○ SHOPPING
○ TRAVEL
○ GOURMET
○ MOVIES
○ MONEY
SET
RETURN
FIG. 6D
ADVERTISEMENT RECEPTION SETTINGS
○ RECEIVE MANUALLY
○ RECEIVE AUTOMATICALLY
○ DO NOT RECEIVE
SET
RETURN
FIG. 6E
GROUP MENU
JOIN GROUP
RETURN
FIG. 6F
GROUP LIST
GAMER ROOM
XX FAN CLUB
DETAIL
RETURN
FIG. 6G
GROUP DETAIL
GAMER ROOM
GAME FANS WANTED
WELCOME
REGARDLESS OF GENRE
REGISTER
RETURN
FIG. 6H
BROADCAST RECEPTION SETTINGS
○ RECEIVE ALL CHANNELS
○ SPECIFY TYPE
○ RECEIVE NONE
SET
RETURN
TO FIG. 6C
FIG. 6I
PUSH MESSAGE RECEPTION SETTINGS
○ RECEIVE ALL MESSAGES
○ SPECIFY TYPE
○ RECEIVE ONLY PERSONAL SPECIFIED
○ RECEIVE NONE
SET
RETURN
TO FIG. 6C

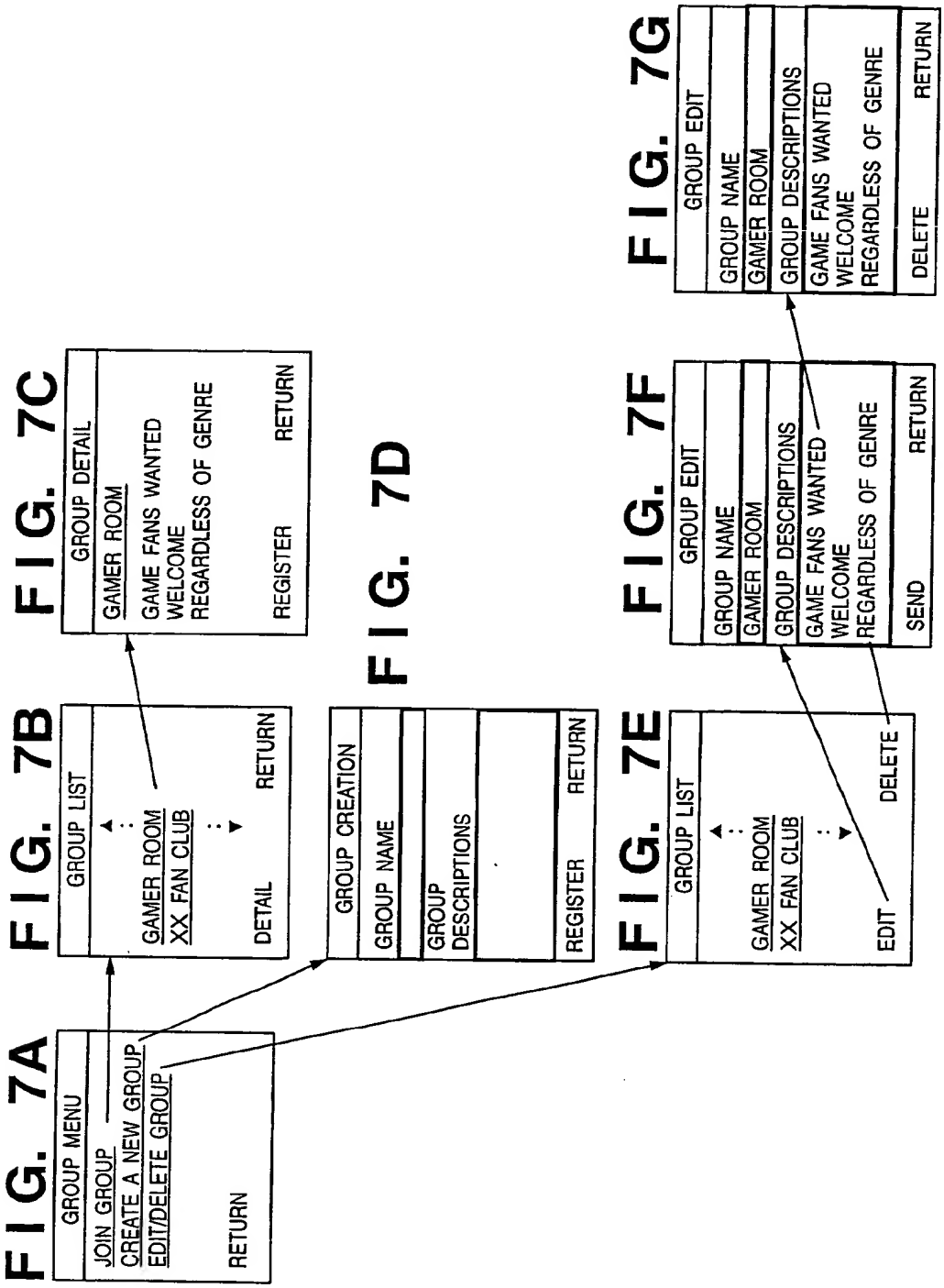


FIG. 7A
GROUP MENU
JOIN GROUP
CREATE A NEW GROUP
EDIT/DELETE GROUP
RETURN
FIG. 7B
GROUP LIST
GAMER ROOM
XX FAN CLUB
DETAIL
RETURN
FIG. 7C
GROUP DETAIL
GAMER ROOM
GAME FANS WANTED
WELCOME
REGARDLESS OF GENRE
REGISTER
RETURN
FIG. 7D
GROUP CREATION
GROUP NAME
GROUP DESCRIPTIONS
REGISTER
RETURN
FIG. 7E
GROUP LIST
GAMER ROOM
XX FAN CLUB
EDIT
DELETE
FIG. 7F
GROUP EDIT
GROUP NAME
GAMER ROOM
GROUP DESCRIPTIONS
GAME FANS WANTED
WELCOME
REGARDLESS OF GENRE
SEND
RETURN
FIG. 7G
GROUP EDIT
GROUP NAME
GAMER ROOM
GROUP DESCRIPTIONS
GAME FANS WANTED
WELCOME
REGARDLESS OF GENRE
DELETE
RETURN

FIG. 8A

FIG. 8B

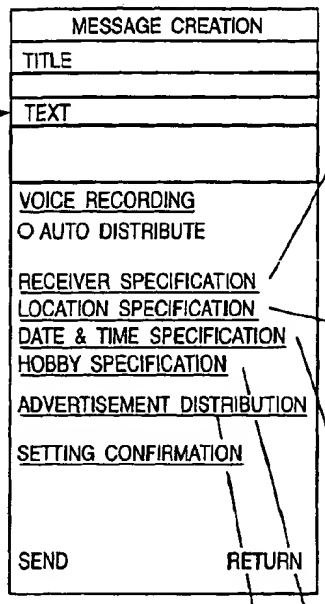


FIG. 8C

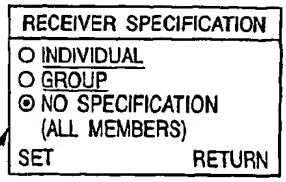


FIG. 8D

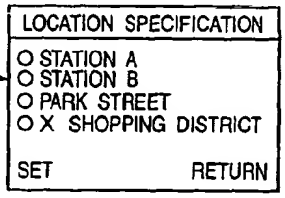


FIG. 8E

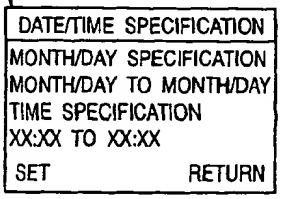


FIG. 8F

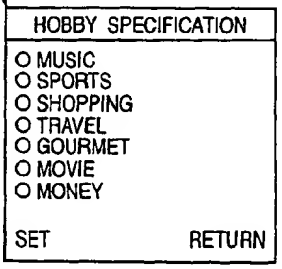


FIG. 8G

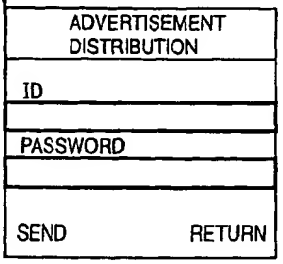


FIG. 9

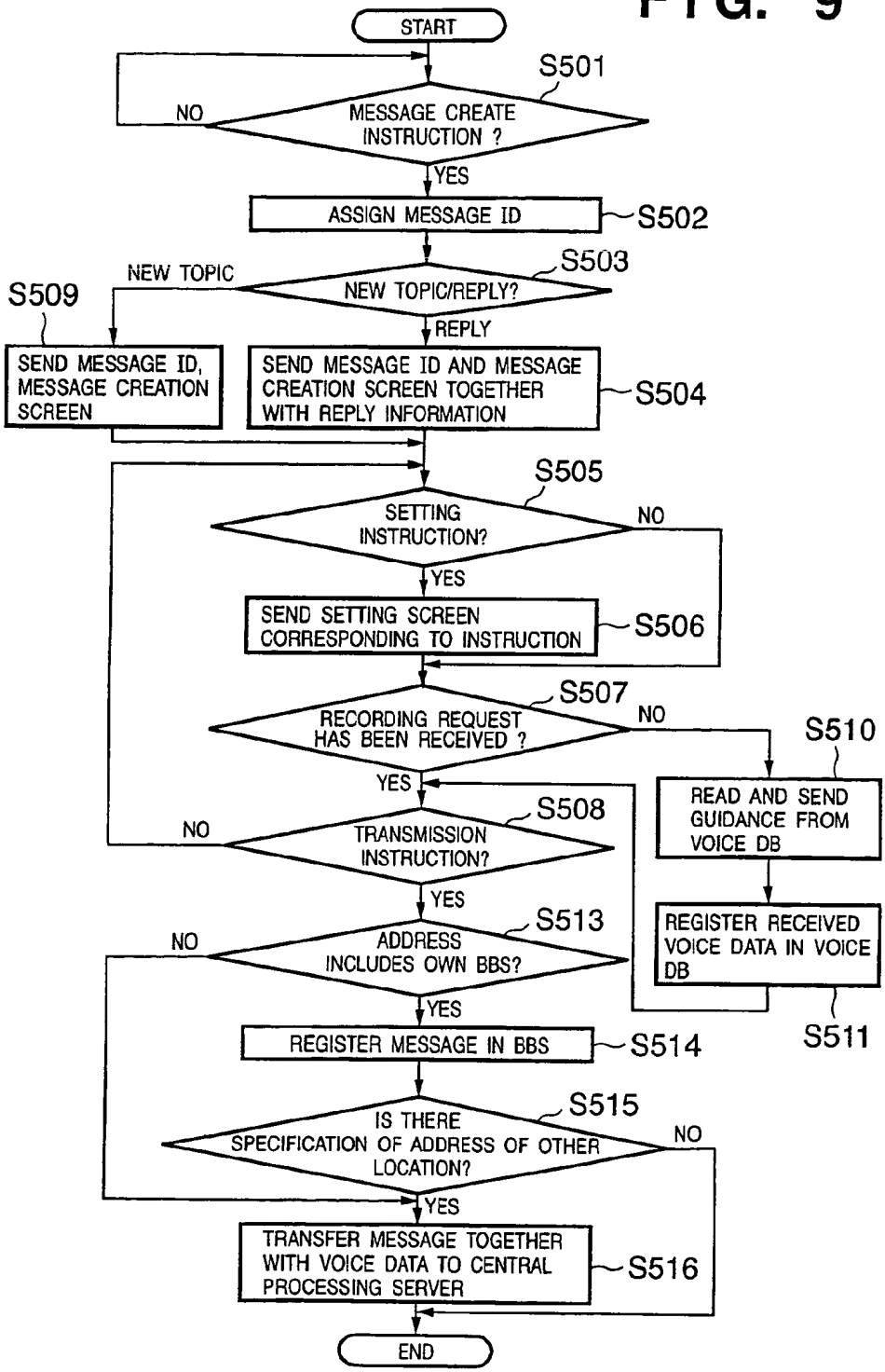
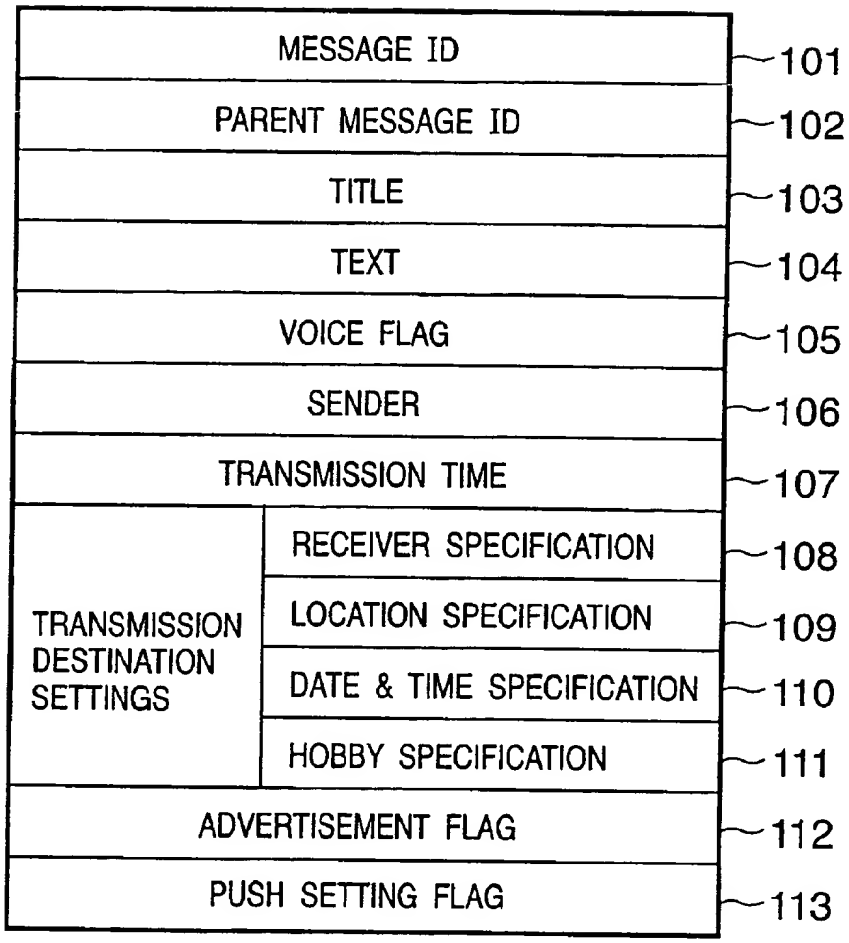

START
S501
MESSAGE CREATE INSTRUCTION ?
NO
YES
ASSIGN MESSAGE ID
S502
S503
NEW TOPIC/REPLY?
NEW TOPIC
REPLY
S509
SEND MESSAGE ID, MESSAGE CREATION SCREEN
SEND MESSAGE ID AND MESSAGE CREATION SCREEN TOGETHER WITH REPLY INFORMATION
S504
S505
SETTING INSTRUCTION?
NO
YES
SEND SETTING SCREEN CORRESPONDING TO INSTRUCTION
S506
S507
RECORDING REQUEST HAS BEEN RECEIVED ?
NO
S510
YES
READ AND SEND GUIDANCE FROM VOICE DB
S508
TRANSMISSION INSTRUCTION?
NO
YES
S513
ADDRESS INCLUDES OWN BBS?
NO
YES
REGISTER RECEIVED VOICE DATA IN VOICE DB
S511
REGISTER MESSAGE IN BBS
S514
S515
IS THERE SPECIFICATION OF ADDRESS OF OTHER LOCATION?
NO
YES
TRANSFER MESSAGE TOGETHER WITH VOICE DATA TO CENTRAL PROCESSING SERVER
S516
END

FIG. 10



MESSAGE DATA FORMAT

FIG. 11A

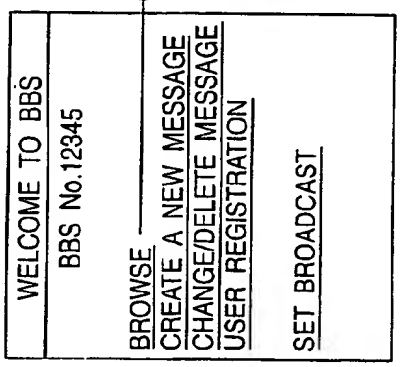


FIG. 11B

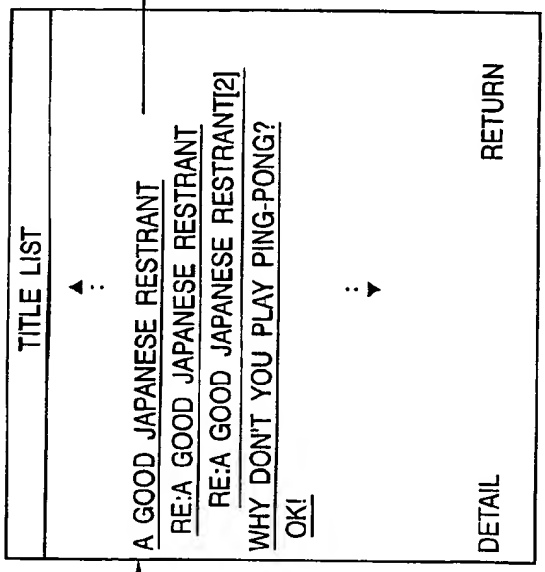


FIG. 11C

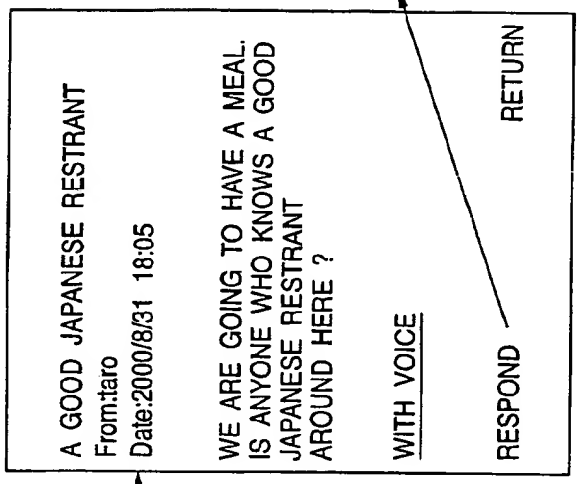


FIG. 12

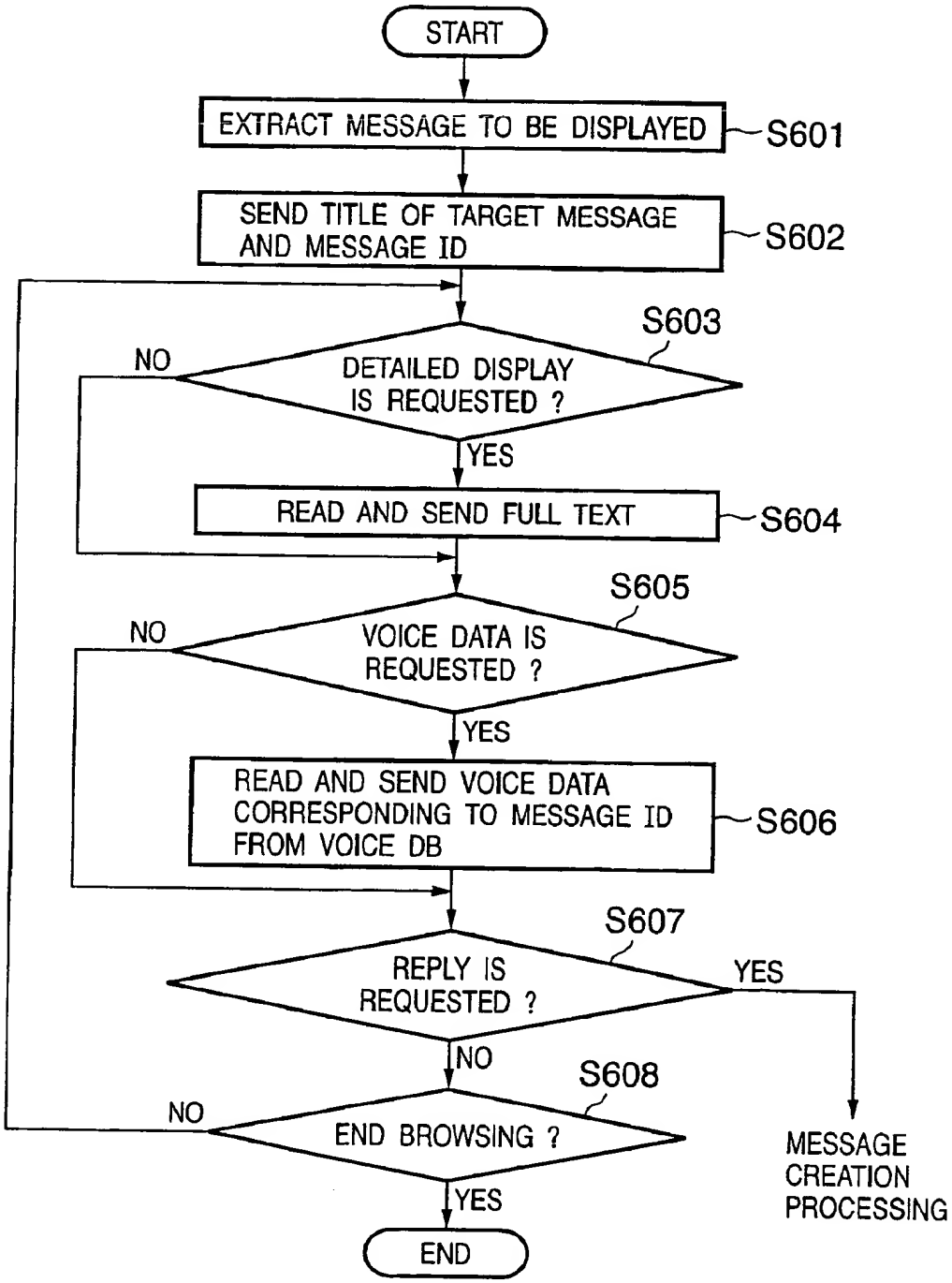

START
EXTRACT MESSAGE TO BE DISPLAYED
S601
SEND TITLE OF TARGET MESSAGE AND MESSAGE ID
S602
S603
DETAILED DISPLAY IS REQUESTED ?
NO
YES
READ AND SEND FULL TEXT
S604
S605
VOICE DATA IS REQUESTED ?
NO
YES
READ AND SEND VOICE DATA CORRESPONDING TO MESSAGE ID FROM VOICE DB
S606
S607
REPLY IS REQUESTED ?
YES
NO
S608
END BROWSING ?
NO
YES
END
MESSAGE CREATION PROCESSING

FIG. 13A

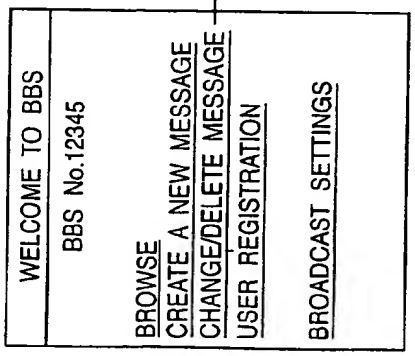


FIG. 13B

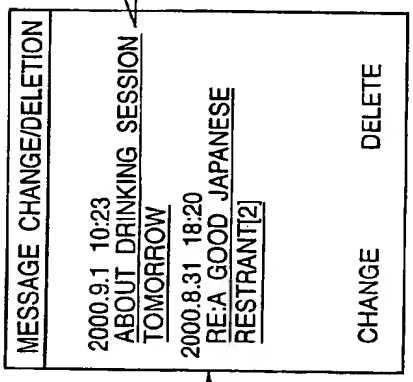


FIG. 13C

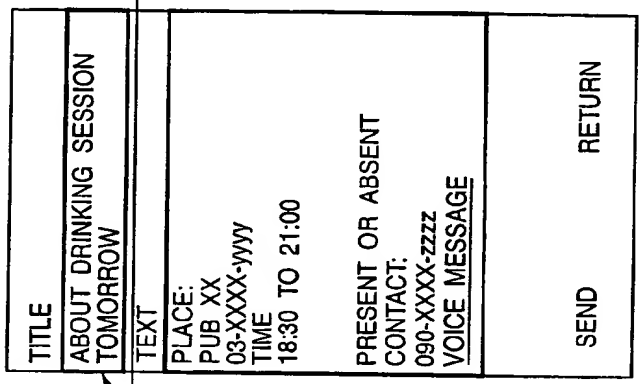


FIG. 13D

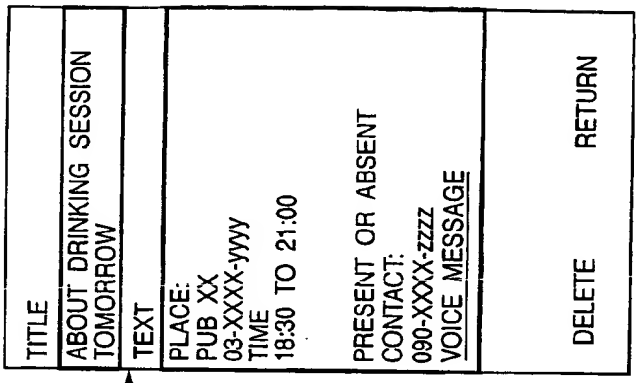


FIG. 14

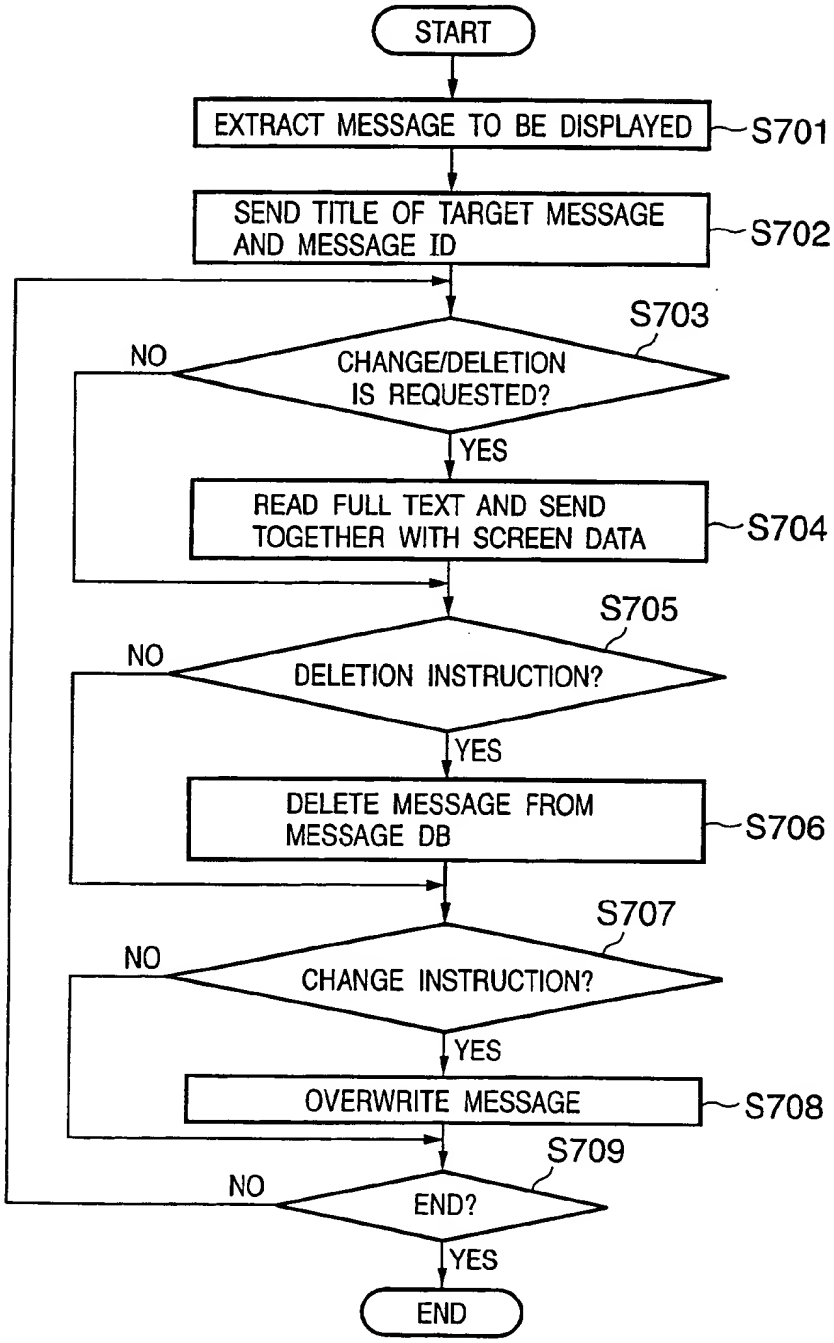
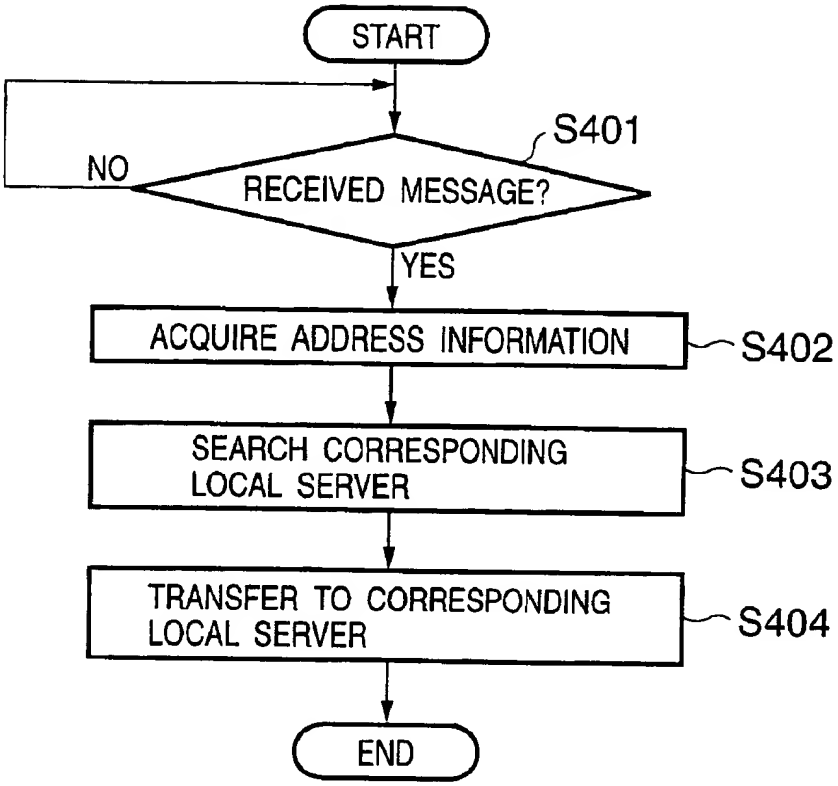

START
EXTRACT MESSAGE TO BE DISPLAYED
S701
SEND TITLE OF TARGET MESSAGE AND MESSAGE ID
S702
S703
NO
CHANGE/DELETION IS REQUESTED?
YES
READ FULL TEXT AND SEND TOGETHER WITH SCREEN DATA
S704
S705
NO
DELETION INSTRUCTION?
YES
DELETE MESSAGE FROM MESSAGE DB
S706
S707
NO
CHANGE INSTRUCTION?
YES
OVERWRITE MESSAGE
S708
S709
NO
END?
YES
END

FIG. 15


START
S401
NO
RECEIVED MESSAGE?
YES
ACQUIRE ADDRESS INFORMATION
S402
SEARCH CORRESPONDING LOCAL SERVER
S403
TRANSFER TO CORRESPONDING LOCAL SERVER
S404
END

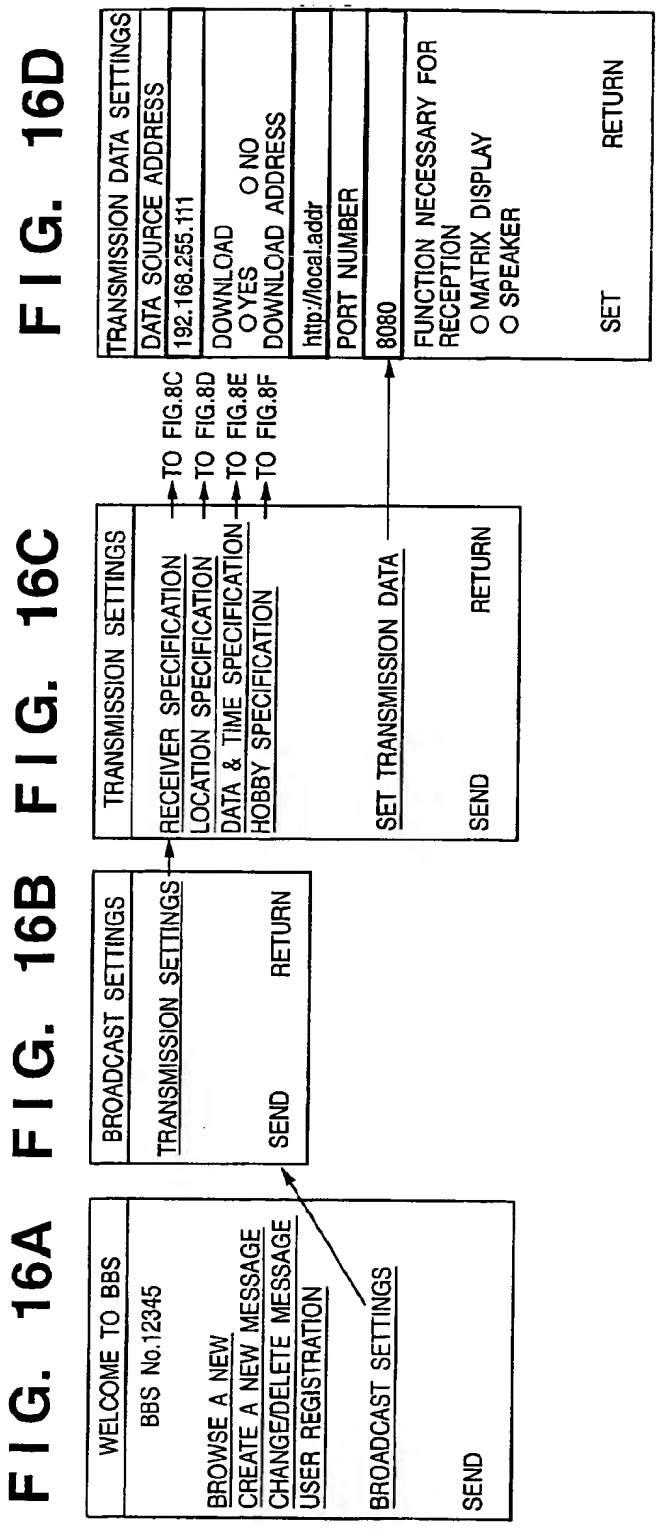


FIG. 16A FIG. 16B FIG. 16C FIG. 16D

FIG. 17A

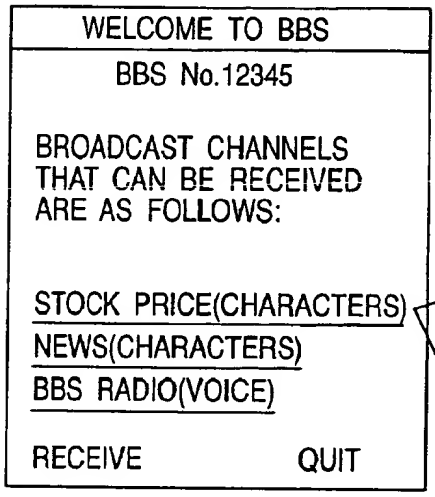


FIG. 17B

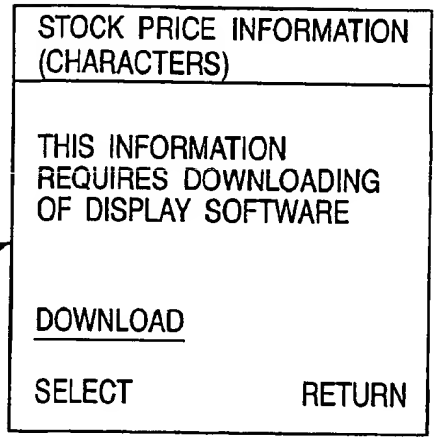


FIG. 17C

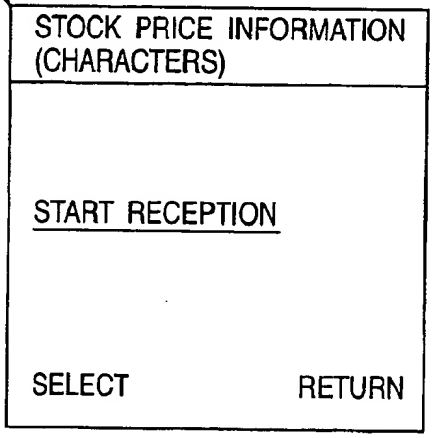
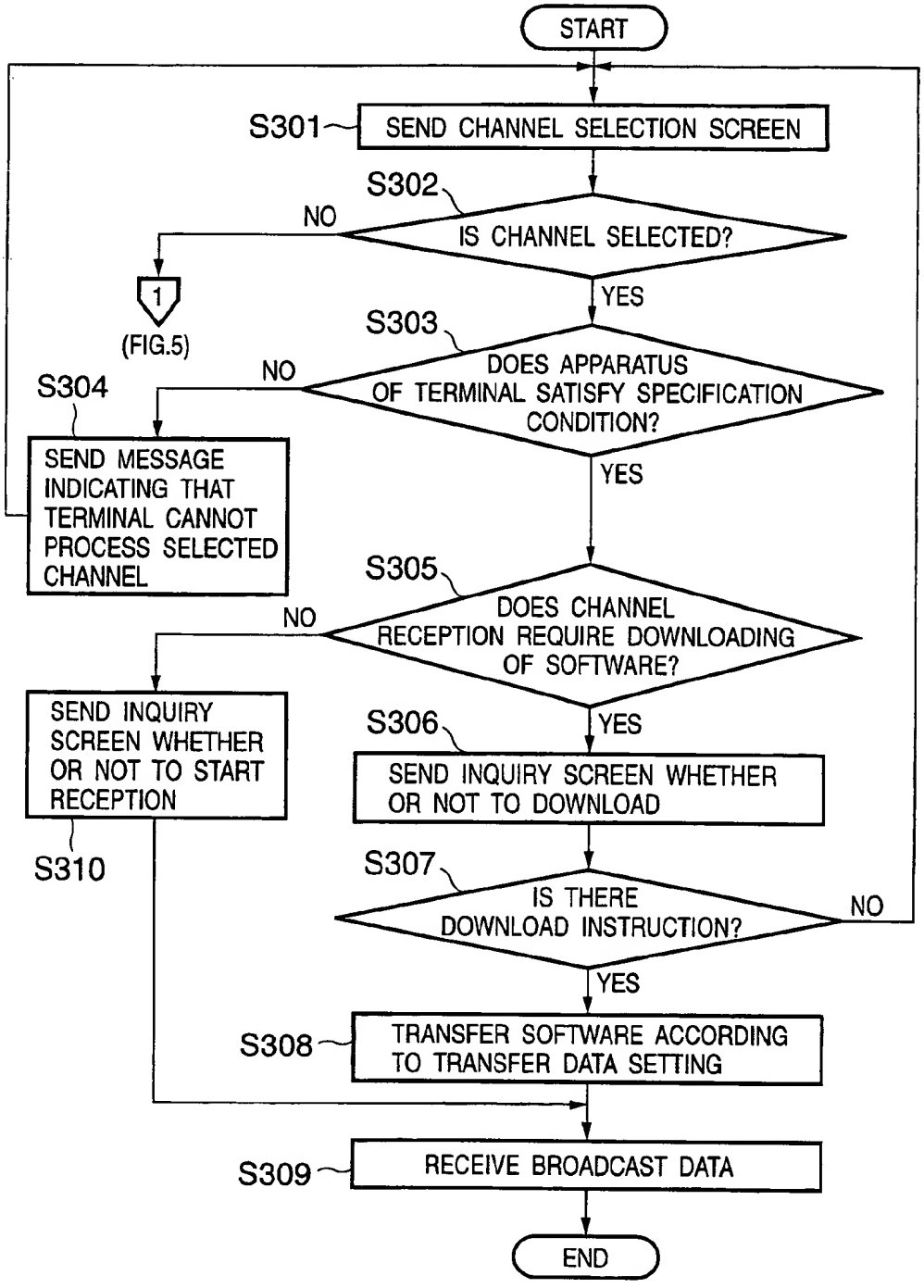


FIG. 18


START
S301
SEND CHANNEL SELECTION SCREEN
S302
IS CHANNEL SELECTED?
NO
YES
1
(FIG.5)
S303
DOES APPARATUS OF TERMINAL SATISFY SPECIFICATION CONDITION?
NO
YES
S304
SEND MESSAGE INDICATING THAT TERMINAL CANNOT PROCESS SELECTED CHANNEL
S305
DOES CHANNEL RECEPTION REQUIRE DOWNLOADING OF SOFTWARE?
NO
YES
SEND INQUIRY SCREEN WHETHER OR NOT TO START RECEPTION
S310
S306
SEND INQUIRY SCREEN WHETHER OR NOT TO DOWNLOAD
S307
IS THERE DOWNLOAD INSTRUCTION?
NO
YES
S308
TRANSFER SOFTWARE ACCORDING TO TRANSFER DATA SETTING
S309
RECEIVE BROADCAST DATA
END

INFORMATION SUPPLY SYSTEM AND CONTROL METHOD THEREOF

FIELD OF THE INVENTION

[0001] The present invention relates to an information supply system and its control method, and more particularly, to an information supply system and its control method capable of selecting a transmission destination of information or information to be acquired.

BACKGROUND OF THE INVENTION

[0002] With progress of communication technologies in recent years, information is now obtainable worldwide without any time differences, and with the spread of the Internet it is now becoming possible to easily access a huge amount of information.

[0003] Generally, to find desired information from homepages on the Internet, the specific addresses of homepages where the desired information is provided should be found magazines, etc., and then access the homepages by using the found addresses, or find homepages where the desired information may be provided by searching a search site which is a homepage dedicated to homepage search, using keyword(s), or track hierarchical links provided by the search site.

[0004] However, in case of a keyword search, the search result is influenced greatly by how a keyword(s) is selected. Moreover, few keyword results in huge number of hits include many unrelated homepages. Therefore, it is necessary add another keyword(s) to reduce undesired (unrelated) homepages, or access possible homepages one by one relying on short summary of pages and check their contents.

[0005] Moreover, at the time of searching information by tracing links for each category provided by the search site, though the possibility that totally irrelevant home pages will be included in links is small, homepages not linked by the search site are not displayed. Moreover, the desired category may not exist, either.

[0006] In addition, information supplied by a homepage searched in this way also includes obsolete information without any information values. Moreover, such a page may not actually exist even if the homepage is accessed.

[0007] Furthermore, since the situation of a searcher, that is, his/her current address, sex, age, etc. have nothing to do with search results, even simply searching information on the neighborhood of the searcher's address also requires the searcher to specify the location by entering a keyword, etc.

[0008] On the other hand, from the standpoint of an information provider, supplying information to those who do not need the information is meaningless. For example, in case of providing information on a bargain sale of children's clothing, supplying the information to childless people is unlikely to bring expected advertisement effects.

SUMMARY OF THE INVENTION

[0009] The present invention has been implemented taking into account the points described above and it is an object of the present invention to provide an information supply system and its control method that supplies desired information to users, capable of automatically selecting and supplying information requested by the users from the whole information.

[0010] Another object of the present invention is to provide an information supply system and its control method that supplies desired information to users, allowing the sender of information to select the destination of the information transmitted using user-specific information.

[0011] That is, according to an aspect of the present invention, there is provided an information system that supplies pre-stored information to service a subscriber terminal that exist in a service area, comprising: at least one local server means having a service area of a predetermined range and a central server means connecting local server means; an information supply system comprising information database means for storing information associated with transmission destination specification conditions, subscriber database means for storing information reception conditions set for each service subscriber terminal, information selecting means for comparing information reception conditions corresponding to service subscriber terminal that exist in the service area of the local server means with transmission destination specification conditions using the information database means and the subscriber database means and selecting only information having transmission destination specification conditions that meet information reception conditions, and information supplying means for presenting only information selected by the information selecting means to the service subscriber terminal that exist in the service area.

[0012] According to an aspect of the present invention, there is provided a method of controlling an information supply system that supplies pre-stored information to service subscriber terminals that exist in a service area, comprising at least one local server means having a service area of a predetermined range, a central server means connecting local server means, and information database means for storing information associated with transmission destination specification conditions and subscriber database means for storing information reception conditions set for each service subscriber terminal; a method of controlling an information supply system comprising an information selecting step of comparing information reception conditions corresponding to service subscriber terminal that exist in the service area of the local server means with transmission destination specification conditions using the information database means and the subscriber database means and selecting only information having transmission destination specification conditions that meet information reception conditions, and an information supplying step of presenting only information selected by the information selecting means to the service subscriber terminal that exist in the service area.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] **FIG. 1** illustrates an overall configuration example of an information supply system according to an embodiment of the present invention;

[0014] **FIG. 2** is a block diagram showing a configuration example of a local server in **FIG. 1**;

[0015] **FIG. 3** is a block diagram showing a configuration example of a central processing server **1** in **FIG. 1**;

[0016] FIG. 4 is a flow chart showing channel control processing of access points in FIG. 1;

[0017] FIG. 5 is a flow chart to explain a flow of entire information supply processing of the local server;

[0018] FIGS. 6A to 6G illustrates a screen display example of mobile radio communication terminal 7 at the time of new registration;

[0019] FIGS. 7A to 7G illustrate group processing menus displayed on a terminal of a registered user and a screen display example related to processing performed from this menu;

[0020] FIGS. 8A to 8G illustrate screen display examples when a message is created on a mobile radio communication terminal 7;

[0021] FIG. 9 is a flow chart to explain processing of the local server when a message is created;

[0022] FIG. 10 illustrates a data format example of a message used in this system;

[0023] FIGS. 11A to 11C illustrate screen display examples during message browsing in the mobile radio communication terminal 7;

[0024] FIG. 12 is a flow chart to explain processing of the local server during message browsing;

[0025] FIGS. 13A to 13D illustrate screen display examples when a message is changed/deleted in the mobile radio communication terminal 7;

[0026] FIG. 14 is a flow chart to explain processing of the local server when a message is changed/deleted;

[0027] FIG. 15 is a flow chart to explain message transfer processing in a central processing server 1;

[0028] FIGS. 16A to 16D illustrate screen display examples in the mobile radio communication terminal 7 when broadcast data is set;

[0029] FIGS. 17A to 17C illustrate screen examples displayed in the mobile radio communication terminal 7 in a service area in which broadcast data exists; and

[0030] FIG. 18 is a flow chart to explain broadcast processing in the local server.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0031] With reference now to the attached drawings, preferred embodiments of the present invention will be explained in detail below.

[0032] (Overall configuration)

[0033] FIG. 1 illustrates an overall configuration example of an information supply system according to an embodiment of the present invention. In the drawing, the information supply system includes three local servers 4 to 6, access points (APs) A-1 to A-3, B-1 to B-3, C-1 to C-3 connected to these local servers respectively and a central processing server 1 to which all local servers 4 to 6 are connected. The central processing server 1 has a function of communication interface between this system and the Internet 11 and supports communications between an information supply server 3 connected to the Internet 11 and the local servers 4 to 6. The central processing server 1 also performs processing such as transferring messages received from the local servers 4 to 6.

[0034] The local servers 4 to 6 are located geographically far from each other having as their service areas zone A to zone C, respectively. Each AP is a radio base station having a service area of a relatively small range, for example a range of less than 100 m. Therefore, the zone covered by each local server is a relatively small range and more specifically, local servers are located at each station, each floor of buildings or each shopping district, etc.

[0035] A mobile radio communication terminal 7 is provided with a function of establishing a radio connection with each AP and when the mobile radio communication terminal 7 enters the support range of each AP, a radio communication channel is established between the AP and mobile radio communication terminal 7.

[0036] Furthermore, the central processing server 1 and local servers 4 to 6 each have user DB2, 8 to 10 having information on registered users of this information supply system. Each user DB has the same contents and synchronization processing is periodically carried out on all user DBs. Of course, it is also possible to provide one user DB accessible to the local servers 4 to 6 and the central processing server with and control the user DB in a centralized manner.

[0037] FIG. 1 shows the case where three local servers are connected to the central processing server 1 and three APs are connected to each local server, but these numbers can be set arbitrarily and the number of APs connected to local servers can be mutually different and set arbitrarily. Likewise, the number and locations of mobile radio communication terminals 7 can be set arbitrarily. (Configuration of a local server)

[0038] FIG. 2 is a block diagram showing a configuration example of the local server 4. The local server 4 includes a CPU 51 that controls the entire system, a ROM 52 that stores programs executed by the CPU 51 and various data, a voice DB 53 that stores or accumulates voice data such as voice guidance and recorded voice, a network I/F 54, which is an interface for data communications between the central processing server 1 and each AP, a RAM 55 used as a work area, etc. of the CPU 51, a user DB 8 that stores information on registered subscribers of this system, a message DB 57 that stores messages registered in an electronic bulletin board service (BBS) serviced by the local server 4 and an HDD 58 that stores programs executed by the CPU 51 and application programs, etc. to be uploaded to mobile radio communication terminals as required.

[0039] These components are connected to each other via a bus (includes a data bus, an address bus and a control bus) of the CPU 51. The voice DB 53, user DB 8 and message DB 57 are described separately in the figure, but these databases can also be configured to occupy some areas of the HDD 58.

[0040] Furthermore, FIG. 2 describes only the local server 4, but the local servers 5 and 6 also have the same configuration and individual explanations of these local servers are omitted here. These local servers 4 to 6 can be implemented by a general-purpose computer apparatus having a network interface.

[0041] Configuration of a Central Processing Server **1**

[0042] **FIG. 3** is a block diagram showing a configuration example of the central processing server **1**. The central processing server **1** includes a CPU **61** that controls the entire system, a ROM **62** that stores programs executed by the CPU **61** and various data, a local server DB **63** that stores the correspondence between locations specified as the transmission destination of messages, etc. and the local servers **4** to **6**, a network I/F **64**, which is an interface for data communications between the Internet **11** and the local servers **4** to **6**, a RAM **65** used as a work area, etc. of the CPU **61**, a user DB **2** that stores information on registered subscribers of this system, and an HDD **68** that stores programs executed by the CPU **61**.

[0043] These components are connected to each other via a bus (data bus, address bus and control bus) of the CPU **61**. The local server DB **63** and user DB **2** are described separately in the figure, but these databases can be configured to occupy some areas of the HDD **68**. This central processing server **1** can be implemented by a general-purpose computer apparatus having a network interface.

[0044] Explanation of Operation

[0045] Hereinafter, how this system supplies information to a subscriber having a mobile radio communication terminal **7** will be explained step by step.

[0046] As described above, a service area (zone) assigned to each local server in this embodiment is relatively small.

[0047] Therefore, it is possible to selectively supply information to the mobile radio communication terminal **7** that exists in the zone. In this embodiment, information is supplied to the mobile radio communication terminal **7** based on an electronic bulletin board service (so-called BBS) and this is combined with a push service that automatically distributes information and a broadcast service that supplies information continuously.

[0048] And, the information supply system according to the present invention is characterized in that only appropriate information is selectively supplied from the information stored in the local server by combining personal information such as the position of a registered subscriber who receives information, previously registered hobby, age and sex, etc.

[0049] For example, the local server A (**4**) having zone A as its service area is provided with information on zone A or its surroundings or BBS that stores only messages specified to be distributed to the mobile radio communication terminal **7** in zone A. Then, the local server A selects and supplies only appropriate information from the information stored in the BBS consulting the user DB **8** on the previously registered personal information about the subscriber of the mobile radio communication terminal **7** that exists in zone A.

[0050] In this embodiment, using information supply service provided by this system requires the user to be registered beforehand. The user is registered by the user's mobile radio communication terminal **7** communicating with the local server via one of APs included in this system. First, it is necessary to establish a channel for a communication between the mobile radio communication terminal **7** and the local server.

[0051] Connection Control Processing

[0052] **FIG. 4** is a flow chart to explain channel control processing carried out by each AP. Each access point sends an inquiry signal once every predetermined time (e.g., every one second) via an antenna (step **S101**). When the mobile radio communication terminal **7** receives the inquiry signal sent by the access point, it responds to the inquiry. When the AP detects the response instep **S102**, the AP assigns the address of an empty channel to the mobile radio communication terminal **7** and sends the assigned address(step **S103**). The AP sends information such as a cryptographic key used for subsequent communications to the terminal **7** using the assigned channel and establishes a communication channel (step **S104**).

[0053] Once the channel is established, the AP acquires subscriber information such as the subscriber number of the mobile radio communication terminal **7** and sends the information to the connected local server (step **S105**). Thereafter, the local server supplies services to the mobile radio communication terminal **7**.

[0054] When the local server detects that there is no response from the mobile radio communication terminal **7** for a predetermined time, for example, in the case where the mobile radio communication terminal **7** moves and goes out of the range covered by the AP, the local server sends a channel disconnection instruction the to the AP. Upon receipt of this channel disconnection instruction (step **S106**), the AP releases the channel assigned to the mobile radio communication terminal **7** (step **S107**) and disconnects the channel (step **S108**).

[0055] In the case where the mobile radio communication terminal **7** moves from an AP to another AP connected to the local server, for example, from the service range of AP (A-1) to the service range of AP (A-2) in **FIG. 1**, the local server **4** can detect establishment of a channel between AP (A-2) at the transfer destination and the mobile radio communication terminal **7**, and in this case, the local sever **4** sends a channel disconnection instruction toAP (A-1) even before instructing channel disconnection due to the presence of the no response time above.

[0056] The interface between the AP and mobile radio communication terminal **7** can be anything if it can implement a relatively narrow range as an AP service range, but it is preferable to be an interface to which protocol supporting both data communication and voice communication is applicable and it is preferable to be an interface with small power consumption because it is mounted in the mobile radio communication terminal **7**.

[0057] As such a radio interface, for example, PHS or Bluetooth can be used. In the case where Bluetooth is used, the service range of each AP is a radius of about 100 m to 10 m. In this case, the mobile radio communication terminal **7** can use a terminal with a configuration with an interface and circuit, etc. to use Bluetooth added to a telephone terminal suited to normal PDC system, etc.

[0058] Information Supply Processing

[0059] As described above, once a radio channel is established between the mobile radio communication terminal **7** and the AP, the AP sends information on the mobile radio communication terminal **7** with which the channel has been

established to the host local server (**FIG. 4**, step **S105**) In response to this, the local server performs processing such as information supply on the mobile radio communication terminal **7**.

[**0060**] As described above, in this embodiment, information is basically supplied to the mobile radio communication terminal **7** through bulletin board services and a push service that automatically distributes information and a broadcast service that supplies information continuously are combined. The communication between the mobile radio communication terminal **7** and the local server is a data communication and, as will be described later, voice data is communicated in packet format as other data.

[**0061**] Furthermore, in this embodiment, a bi-directional communication between the mobile radio communication terminal **7** and local server is basically implemented by the local server sending display data such as a menu screen to the mobile radio communication terminal **7** and the mobile radio communication terminal **7** selecting links included in the display screen or sending an instruction for sending data after inputting the data input area and selected area included in the display data and thereby sending data to the local server.

[**0062**] Then, the information supply processing of the local server in this embodiment will be explained using a flow chart shown in **FIG. 5** and taking the operation of the local server **4** as an example.

[**0063**] First, in step **S201**, the subscriber information, for example, the subscriber number of the mobile radio communication terminal **7** is received from the AP described above. Then, the user DB **8** is searched by using this subscriber number and it is checked whether the subscriber is an already registered user (service subscriber) or not (step **S202**).

[**0064**] In the case where the check result shows that the subscriber is an unregistered user, the service initial screen display data of BBS is sent (step **S208**) and processing is performed according to the instruction of the terminal thereafter (step **S209**). Registration of the user will be described later.

[**0065**] On the other hand, in step **S202**, in the case where the subscriber is confirmed to be a registered user, it is checked whether there is any message to be pushed with reference to the user DB **8** and message DB **57** or not (step **S204**). More specifically, of the messages stored in the message DB **57**, there are checked whether there is any message in which the subscriber of the mobile radio communication terminal **7** is set as the transmission destination and in which a "Push" is set. Settings referenced here such as setting of the transmission destination for each message will be described later.

[**0066**] As described above, "Push" means a service that the system forcibly sends information even if the subscriber does not access the information. Therefore, if a "Push" message exists, first the screen display data that lists the message (title) is sent to the mobile radio communication terminal **7** (step **S205**).

[**0067**] After the push message is sent, if it is detected that processing about the push message has completed, for example, an instruction for returning to the initial screen, it is checked whether there is service subscriber broadcast data or not (step **S206**). The broadcast data refers to data substantially and continuously supplied, for example data displayed on a character news display apparatus placed in the street.

[**0068**] In step **S206**, as in case of the check in step **S204**, it is determined whether there is broadcast data that can be supplied for each registered user with reference to the setting contents about broadcast data reception of the registered user, user information registered in the user DB **8** such as personal information and the setting contents of the user who is the transmission target set by the supplier of the broadcast data.

[**0069**] In the case where broadcast data that can be supplied exists, in step **S207**, screen display data indicating that the broadcast data exists is sent and then the broadcast data is supplied under instructions from the user. Details of the broadcast processing will be described later.

[**0070**] When the broadcast data processing is completed in such a case where a broadcast data reception completion instruction is received or where a response of not receiving the broadcast data is received, data for displaying the service initial screen is sent (step **S208**). Thereafter, processing is performed under instructions from the terminal (step **S209**).

[**0071**] Of course, the process may move to steps **S203**, **S204**, **S206**, etc. depending on the instructions received through processing in step **S209**.

[**0072**] In this way, the local server **4** consults the transmission destination specification information registered in a message (or broadcast data) and the reception data setting registered by the service subscriber and supplies only the data in which these two match to the service subscriber of the mobile radio communication terminal **7** that exists in the service area of the AP (**A-1** to **A-3**).

[**0073**] User Registration Processing

[**0074**] Specific user registration processing will be explained using **FIGS. 6A** to **6G** below.

[**0075**] The user registration processing is performed-by selecting the "User registration" menu included in the BBS initial screen shown in step **S208** in **FIG. 5**.

[**0076**] **FIG. 6A** shows an example of the BBS initial screen sent from the local server to the user terminal. In **FIGS. 6A** to **6G**, underlined characters (lines) indicate links. By selecting the line and pressing a given key of the terminal assigned, for example, to "Send" displayed on the last line of the menu, a request for the display data of another menu screen (submenu) associated with the link will be sent to the local server and the local server will send the display data corresponding to the request, then another menu screen will be displayed.

[**0077**] That is, in **FIG. 6A**, if the user selects "User registration" on the terminal screen and presses the key assigned to "Send", a request for sending the user registration screen is sent to the local server, and if the terminal side interprets and displays the user registration screen display data that the local server sends in response to this request, the user registration screen as shown in **FIG. 6B** is displayed.

[0078] Such a communication between the mobile radio communication terminal **7** and local server can be performed according to, for example, an HTTP protocol. In this case, as the language that the local server sends to the mobile radio communication terminal **7** and that can be interpreted on the terminal side, HTML or its instruction-extended versions can be used.

[0079] The user registration screen according to this embodiment includes the following fields:

[0080] 1) Handle name setting field

[0081] Field to set a handle name (pen name) displayed on a message contributed to BBS. Enter characters.

[0082] 2) Age setting field

[0083] Field to set user's age. Enter a number.

[0084] 3) Sex selection field

[0085] Field to select user's sex. Select either a male or female radio button.

[0086] 4) e-mail address setting field

[0087] Field to set user's e-mail. Enter alphanumeric characters.

[0088] 5) Phone number setting field

[0089] Field to enter user's subscriber number. It is also possible to display the user's subscriber number notified from the AP as default setting.

[0090] 6) Link to hobby settings screen

[0091] Link to screen to set fields of interest for user

[0092] 7) Link to advertisement reception settings screen

[0093] Link to screen to set whether or not to receive advertisement message or set reception method

[0094] 8) Link to group settings screen

[0095] Link to group setting screen such as join a registered group(s) and creation of new group, etc.

[0096] 9) Link to broadcast reception settings screen

[0097] Link to broadcast reception setting screen to set whether broadcast data is received or not or type of reception, etc.

[0098] 10) Link to push message reception settings screen

[0099] Link to push message reception setting screen to set whether message with push setting (auto-distribution setting) is received or not or type of reception, etc.

[0100] 11) Link to registration confirmation screen

[0101] Link to be selected when final registration is performed. Link to final confirmation screen It is optional for the user to decide which of these setting fields should be set as mandatory items, but at least the handle name, phone number and e-mail address must be included in the mandatory items.

[0102] Hobby Settings

[0103] When the user performs hobby setting, the user selects the hobby setting link from the user registration screen and presses the key assigned to "Send" on the terminal. This makes the hobby setting screen (**FIG. 6C**) appear on the terminal as in the case of a transmission request or response of the above-described user registration screen display data.

[0104] The hobby setting screen is a screen to set the category of information that the user wants to receive or see and show a list of several roughly selected categories as shown in **FIG. 6C**. Each item in the list has a radio button and when the user selects an arbitrary number of radio buttons corresponding to categories of interest and presses the terminal key corresponding to "Set", the setting contents is temporarily stored in the terminal and the user registration screen (**FIG. 6B**) is returned to. If the terminal key corresponding to "Return" is pressed instead of "Set", nothing is set and the user registration screen (**FIG. 6B**) is returned to.

[0105] Advertisement Reception Settings

[0106] To perform an advertisement reception setting, the user selects the advertisement reception setting link from the user registration screen and presses the terminal key assigned to "Send". This makes an advertisement reception setting screen (**FIG. 6D**) appear on the terminal as in the case of a transmission request or response of the above-described user registration screen display data.

[0107] The advertisement reception settings screen is a screen to set whether the user wants to have a message, which is set as "Advertisement" (advertisement message) from among the messages stored in BBS, supplied or not or set whether a push service should be validated or not. The advertisement reception setting screen in **FIG. 6D** is provided with the following choices and the user selects and sets one:

[0108] 1) Receive manually

[0109] 2) Receive automatically

[0110] 3) Do not receive

[0111] 1) is a setting that invalidates the push service about the advertisement message and includes an advertisement message (with a push setting) in the message list displayed when the user browses BBS.

[0112] 2) is a setting that validates the push service by the local server for an advertisement message with a push setting and when the user enters the service area of the local server, automatically sends advertisement messages with push settings that match other settings to the mobile radio communication terminal **7**. Of course, advertisement messages without push settings are displayed in the message list displayed when the user browses BBS.

[0113] 3) is set when the user does not want to receive any advertisement message.

[0114] When the user selects one radio button corresponding to the item to be specified and presses the terminal key corresponding to "Set", the setting contents is temporarily stored in the terminal and the user registration screen (**FIG. 6B**) is returned to. If the terminal key corresponding to

"Return" instead of "Set" is pressed, nothing is set and the user registration screen (**FIG. 6B**) is returned to.

[0115] Group Settings-Join a New Group(s))

[0116] When the user carries out a group setting, the user selects a group setting screen link from the user registration screen (**FIG. 6B**) and presses the terminal key assigned to "Send". This makes the group settings screen (**FIG. 6E**) appear on the terminal as in the case of a transmission request or response of the above-described user registration screen display data.

[0117] The group settings screen is a screen used when the user joins a group registered in BBS, creates a new group or edits/deletes a group created by the user.

[0118] In this embodiment, a "group" is a group of registered users and no restrictions by hobby setting item, etc. are imposed when a user joins the group. A group can be specified, for example, as the transmission destination of a message created.

[0119] As shown in **FIG.** 6E, the group menu screen shows only a new joining link for a user to newly joins the group. On the other hand, when a registered user calls up the group setting screen from a predetermined screen, a group creation link to be selected when a new group is created and a group edition/deletion link to be selected when the group created by the user is edited or deleted are displayed in addition to the new joining link as shown in **FIG. 7A**.

[0120] When an unregistered user wants to join a new group, the user selects the new joining link in **FIG. 6E and** a list screen (**FIG. 6F**) to specify a group appears. The group list screen shows a list of all group names registered in the local server as a link.

[0121] Of course, it is also possible to display, not the screen showing the list of all groups, but a group name obtained by narrowing the search range through a screen for searching names in alphabetical order beforehand.

[0122] When the user selects a group name of interest from among group names shown in the group name list and presses the key assigned to "Detail", a predetermined brief introduction to the group (**FIG. 6G**) is displayed. Here, if the user presses the key assigned to "Register", information that the user wants to get registered in the group displayed is temporarily stored in the terminal and the user registration screen (**FIG. 6B**) is returned to.

[0123] In the case where a registered user newly joins the group, as shown in **FIGS. 7B and 7C**, the screen displayed on the terminal is the same as in the case of the unregistered user, but different in that when the key assigned to "Register" in **FIG. 7C** is pressed, a registration instruction is sent to the local server. This is because in the case of an unregistered user, registration itself may be canceled by the time definitive registration is instructed.

[0124] Group Setting-Create a New Group

[0125] Here, other group settings to be carried out by a registered user will also be explained. Selecting the new group creation link ("Create new group") will display the group creation screen (**FIG. 7D**) to enter a group name and group descriptions, which are displayed on the group list display screen and detail display screen (**FIGS. 6F, 6G, 7B** and **7C**).

[0126] When the user enters a group name and group descriptions on this screen and presses the key assigned to "Register", the input data is sent to the local server and registered in the group DB provided in a predetermined area of the HDD **53** together with the subscriber number of the registrant.

[0127] Group Setting-Edit/Delete Settings

[0128] On the other hand, when the link to edit/delete an existing group ("Edit/delete group") included in the group setting screen (**FIG. 7A**) is selected, the local server **4** searches the group DB using the subscriber number of the registered user and transmits the list display data of the group created by the registered user. The registered user selects a desired group to be edited or deleted from this list display screen (**FIG. 7E**) and presses the key assigned to "Edit" or "Delete".

[0129] When "Edit" is instructed (**FIG. 7F**) or when "Delete" is instructed (**FIG. 7G**), the local server **4** reads the name of the specified group and group descriptions from the group DB and sends it together with the display screen data. The group name and group descriptions are displayed on the terminal and in the case of editing, the user edits this data and presses the key assigned to "Send" to send the changed data to the local server. The local server overwrites the group DB with the received data.

[0130] On the other hand, if the key corresponding to "Delete" is pressed in **FIG.** 7G, the local server deletes the record on the relevant group from the group DB.

[0131] Back in **FIGS. 6A** to 6G, if an unregistered user carries out various settings, selects a "Register" link in **FIG. 6B** and presses the key assigned to "Send", when a confirmation message "Do you register?", etc. is displayed and "Register" is instructed again, the mobile radio communication terminal **7** sends the subscriber number and each setting item to the local server. Of course, it is also possible to configure the system so that a list of the setting contents is displayed to the user together with the message "Do you register?".

[0132] Like this, if the user DB is changed as in the case where an unregistered user is newly registered or a registered user has changed the registered contents, etc., the relevant local server notifies the central processing server **1** of the change contents. The central processing server **1** reflects the change contents in its own user DB **2** and at the same time instructs local servers other than the notification source to reflect the change in the user DB owned by each local server. This maintains synchronization between all user DBs.

[0133] Reception Settings of Broadcast Messages

[0134] When the user performs settings regarding reception of broadcast data, the user selects a broadcast reception setting screen link from the user registration screen (**FIG. 6B**) and presses the terminal key assigned to "Send". This makes the broadcast reception settings screen (**FIG. 6H**) appear on the terminal as in the case of a transmission request or response of the above-described user registration screen display data.

[0135] The broadcast reception settings screen is provided with the following three choices and the user can carry out settings by selecting a desired choice and pressing the key corresponding to "Set":

[0136] 1) Receive all channels that can be received

[0137] 2) Receive only specific type of channel

[0138] 3) Receive no broadcast data

[0139] However, if "Specify type" is selected, the above-described hobby settings (**FIG. 6C**) screen is displayed and the setting is completed by specifying a category of the broadcast data that the user wants to receive.

[0140] Reception Settings of Push Message

[0141] In the case where the user carries out settings regarding reception of a push message, the user selects a push message reception settings screen link from the user registration screen (**FIG. 6B**) and presses the terminal key assigned to "Send". This makes a push message reception settings screen (**FIG. 6I**) appear on the terminal as in the case of a transmission request or response of the above-described user registration screen display data.

[0142] The push message reception settings screen is provided with the following four choices and the user can carry out settings by selecting a desired choice and pressing the key corresponding to "Set":

[0143] 1) Receive all push messages

[0144] 2) Receive only a push message of a specific type

[0145] 3) Receive only a push message directed to user

[0146] Personally Specified

[0147] 4) Receive no push message

[0148] Of these four, if the user sets "Receive only personally specified push message", the user will only receive the message for which a specific individual (the user) is set as the transmission destination from among the messages for which a push setting (auto distribution) is set in the message creation processing, which will be described later.

[0149] Furthermore, in the case where "Specify type" is selected, the above-described hobby settings screen (**FIG. 6C**) is displayed and the category of a push message that the user wants to receive is specified and the setting ends.

[0150] Message Creation Processing

[0151] Then, the message creation processing will be explained by using **FIGS. 8A** to **8G** showing screen display examples in the mobile radio communication terminal **7** and **FIG. 9**, which is a flowchart showing processing of the local server. In the following explanation, message creation processing between the mobile radio communication terminal **7** and local server **4** will be described as an example. However, the local servers in this system will also operate in the same way on a message creation request from an apparatus, which can access this system over the Internet

[0152] A message is created in the following cases:

[0153] 1) Create a new topic

[0154] 2) Respond to a message shown during browsing of BBS

[0155] First, creation of a new topic will be explained using the screen display example in **FIGS. 8A** to **8G** and the flow chart in **FIG. 9**.

[0156] When a message creation link is selected from the service initial screen (**FIG. 8A**) (step **S501**), the local server **4** assigns a message ID to the new message (step **S502**). Then, since the message is creation of a new topic (step **S503**), the display data of the message creation screen will be sent together with the message ID (step **S509**).

[0157] As a result, the message creation screen (**FIG. 8B**) is displayed on the terminal.

[0158] The message creation screen is provided with the following:

[0159] 1) Title field

[0160] Character input field to input message title

[0161] 2) Text field

[0162] Character input field to input message text

[0163] 3) Voice recording link

[0164] Link selected to attach voice to message

[0165] 4) Push setting field ("auto distribution")

[0166] Field set to attach push attribute to message created

[0167] 5) Receiver specification screen link

[0168] Link selected to specify receiver of message created by individual or group, etc.

[0169] 6) Location specification screen link

[0170] Link selected to specify receiver of message created by location (local server)

[0171] 7) Date & time specification screen link

[0172] Link selected to only send message created during specific period

[0173] 8) Hobby specification screen link

[0174] Link selected to specify receiver of message created according to hobby setting set by receiver

[0175] 9) Advertisement distribution setting screen link

[0176] Link selected to handle message created as advertisement in system

[0177] 10) Setting confirmation screen link

[0178] Link selected to confirm contents of setting carried out on message created before sending message

[0179] Selection of each link is detected by the local server (step **S505**) and the local server sends the setting screen corresponding to the selected link (step **S506**).

[0180] Creation of Title and Text

[0181] The title and text are input by entering characters from the terminal into each field. The recording link is a link selected to attach voice data to a message and when the local server detects the selection of the recording link (step **S507**), the local server reads a voice guidance to urge recording from voice DB **53** and sends it to the mobile radio communication terminal **7** (step **S510**).

[0182] This voice guidance (packet data) is replayed as a voice signal in the mobile radio communication terminal **7**

and notified to the user. The user records speech from a microphone according to the guidance. This voice is digitized inside the terminal, coded if necessary and sent to the local server **4** as voice packet data. This voice packet data is sent with a message ID and an identifier indicating that the packet is a voice packet included in the header.

[0183] Upon receipt of this voice packet, the local server **4** converts the voice packet to a readable format according to the message ID and registers it in the voice DB **53** (step S511). The mobile radio communication terminal **7** sends voice data and then sets a voice flag (described later) in the message data.

[0184] Various Settings

[0185] The push settings field is selected to carry out auto distribution from the local server to a transmission target subscriber using the above-described push service (select a radio button).

[0186] When the receiver specification screen link is selected, the receiver specification screen in **FIG. 8C** is displayed. The receiver specification screen is provided with an individual specification link, which is selected to specify a specific individual, a group specification link, which is selected to specify a group and a default all-member specification field.

[0187] If the individual specification link ("individual") is selected here, an address setting screen, which is not shown in the drawing, is displayed and the transmission destination is set to a specified individual by inputting information with which the receiver is identifiable such as the receiver's e-mail address, handle name and subscriber number. When individual specification is performed, by combining the "individual" designation with the location specification, which will be described later, such a message will be browsed by the specified individual only when the specified individual enters the service area of the specified local server.

[0188] Furthermore, if the group specification link is selected, a group list screen as shown in **FIG. 6F** is displayed and selecting a desired group from the relevant screen will select the subscriber who belongs to the selected group as the transmission target.

[0189] Furthermore, if the location specification screen link is selected on the message creation screen, the location specification screen shown in **FIG. 8D** is displayed.

[0190] The location specification screen shows a service area of a local server that belongs to this system. It is possible to specify an arbitrary number of service areas and register messages in BBS of another location by specifying the location. In this case, as will be described later, a message is distributed by transferring the message from the local server to the central processing server **1** and by the central processing server **1** judging the local server corresponding to the location specified as the destination of the message.

[0191] Furthermore, when specifying the location, it is also possible to make it selectable whether a message should be registered or not in the BBS supplied by the local server that created the message. It is further possible to specify the location using a number assigned to each BBS (local server). The BBS number is displayed as "BBS No. XXXX" on the initial screen in **FIG. 8A**, for example.

[0192] Selecting the date & time specification screen link shows the date & time specification screen shown in **FIG. 8E**. The date & time specification screen is provided with the month/date input field specified in day units and the hour input field specified in hour and minute units.

[0193] The message with these fields filled with settings is only sent to the users who exist in the service area of the local server (specified local server in the case where the location is specified) in the specified date/time range.

[0194] If the hobby specification link is selected, the hobby specification screen shown in **FIG. 8F** is displayed. This screen is the same as the hobby setting screen during the user setting and if an arbitrary number of categories are selected from the categories displayed on this screen, the subscribers who have selected the selected categories in the hobby setting become the message transmission targets.

[0195] Furthermore, if the advertisement distribution setting screen link is selected, the advertisement distribution authentication screen in **FIG.8G** is displayed. The advertisement distribution authentication screen is provided with an ID input field and password input field where the user ID number and password for the advertisement distribution registered beforehand are entered, respectively. The ID and passwords entered are sent to the local server together with the subscriber number by pressing the key assigned to "Send" and the local server searches the user DB based on the subscriber number and performs authentication against the registered ID and password. If the authentication is performed correctly, a message display data such as "Registered as advertisement message" is sent to the mobile radio communication terminal **7**. Upon receipt of this message, the mobile radio communication terminal **7** sets an advertisement flag (described later) corresponding to the message being created.

[0196] On the message creation screen in **FIG. 8B**, if the key assigned to "Send" is pressed, a message with the configuration as shown in **FIG. 10** is sent to the local server **4**.

[0197] In **FIG. 10**, the message data has the following fields:

[0198] 1) Message ID field **101**

[0199] Field to store message ID assigned from local server **4**

[0200] 2) Parent message ID field **102**

[0201] Field to store message ID of message (parent message), which is link source, in message linked with another message such as reply

[0202] 3) Title field **103**

[0203] Field to store message title

[0204] 4) Text field **104**

[0205] Field to store message text

[0206] 5) Voice flag **105**

[0207] Field to indicate whether message comes with recorded voice or not. This field is set if voice is recorded when message is created.

[0208] 6) Sender field **106**

[0209] Field to store subscriber number of sender when message is sent from a mobile radio communication terminal **7**. When the message is stored in the message DB, the subscriber number is converted to a handle name by the local server using the user DB **8**.

[0210] 7) Transmission time field **107**

[0211] Field to store date/time extracted from a clock inside mobile radio communication terminal **7** when message is sent

[0212] 8) Receiver specification field **108**

[0213] Field to store contents specified on receiver specification screen (**FIG. 8C**) when message is created

[0214] 9) Location specification field **109**

[0215] Field to store contents specified on location specification screen (**FIG. 8D**) when message is created

[0216] 10) Date & time specification field **110**

[0217] Field to store contents specified by date & time specification screen (**FIG. 8E**) when message is created

[0218] 11) Hobby specification field **111**

[0219] Field to store contents specified by hobby specification screen (**FIG. 8F**) when message is created

[0220] 12) Advertisement flag **112**

[0221] Flag set for message correctly authenticated via advertisement distribution setting screen (**FIG. 8G**) when message is created

[0222] 13) Push setting flag **113**

[0223] Flag set for message with push setting on message creation screen (**FIG. 8B**)

[0224] Upon receipt of a message in such a format, the local server **4** first refers to the location specification field **109** and confirms whether the received message should be stored in the message DB **57** or not, that is, whether only other locations are specified or not (step **S513**).

[0225] If specification of only other locations is not the case, that is, when it is decided that the message needs to be registered in the message DB **57**, the received message is registered in the message DB **57** (step **S514**).

[0226] Then, if other locations are specified (step **S515**), the message data is transferred to the central processing server **1**. In this case, if a check of the voice flag **105** in the message shows that voice data is attached to the message data, the voice data is read from the voice DB **53** and transferred to the central processing server **1** together with the message data (step **S516**).

[0227] As will be described later, upon detection of reception of a message, the central processing server **1** checks its location specification field **109** and identifies the local server to which the message is to be transferred using the location specification contents and local server DB **63** (**FIG. 3**). Then, the message is transferred to the specified local server. In the case where a plurality of specified local servers is specified, the message is copied and transferred to each local server.

[0228] Browsing Processing

[0229] Then, browsing processing and creation of a reply message from the browsing processing will be explained by using **FIG. 9**, **FIG. 10** and **FIG. 12**. **FIGS. 11A** to **11C** are terminal screen display examples during browsing and FIG. **12** is a flow chart showing browsing processing in the local server.

[0230] When browsing is performed from the initial screen (**FIG. 11A**), a browsing instruction is sent to the local server **4** by selecting a browsing screen link "Browse" in the initial screen. In response to this instruction, the local server **4** searches the user DB **8** using the subscriber number of the mobile radio communication terminal **7** and compares the setting contents of the relevant subscriber and the setting contents of the message in the message DB, and thereby extracts the message in which both conditions match (step **S601**).

[0231] Then, the local server **4** sends the title and message ID of the message to be displayed to the mobile radio communication terminal **7** (step **S602**). As a result, the message title is displayed on the screen of the mobile radio communication terminal **7** (**FIG. 11B**). As explained by using **FIG. 5**, in the case where there is a push message that matches the user setting in the message DB, when the mobile radio communication terminal **7** enters the service area and a radio channel is established, the same extraction processing as that in step **S601** is performed on the message with a push setting and a title display screen as shown in **FIG. 11B** is displayed.

[0232] When a desired title is selected from the title display screen in **FIG. 11B** and the key assigned to "Detail" is pressed (step **S603**), a detailed display request and message ID of the message requested to be displayed are sent from the mobile radio communication terminal **7** to the local server **4**. Upon receipt of this, the local server **4** searches the message DB **57** using the message ID and sends items necessary for a predetermined detailed display format such as message text, voice flag, transmission date/time, sender handle name, etc. to the mobile radio communication terminal **7** (step **S604**). As a result, the whole text of the message is displayed on the screen of the mobile radio communication terminal **7** (**FIG. 11C**).

[0233] In this embodiment, the title, handle name of the sender, transmission date/time and text are displayed on the whole text display screen shown in **FIG. 11C**, and in the case where message voice recording is linked, a link to the voice data ("With voice") is displayed.

[0234] If the user selects the link to the voice data from the whole message text display screen (step **S605**), the local server **4** reads the voice data having the same ID as the message ID displayed from the voice DB **53** and sends it to the mobile radio communication terminal **7** (step **S606**). The mobile radio communication terminal **7** replays the received voice data and outputs the voice attached to the message through the speaker of the mobile radio communication terminal **7**.

[0235] On the other hand, if the user presses the key assigned to "Respond" from the whole message text display screen (step S607), the process transitions to the message creation processing shown in FIG. 9. In this case, in step S503 in FIG. 9 it is detected that the message to be created is a reply message.

[0236] Then, the local server 4 sends the message ID assigned to the new message in step 502, the display data of the message creation screen shown in FIG. 8B, reply information such as the message ID of the parent message, reply title (for example, the parent message title with "Re:" added) and transmission destination specification information such as location specification information in the parent message (step S504).

[0237] As a result, the mobile radio communication terminal 7 displays a message creation screen with reply information embedded. Of course, in step S504, it is also possible to send only the display data of the message creation screen and message ID and acquire reply information by an application on the mobile radio communication terminal 7 side. Since the subsequent processing in creation of a reply message (in and after step S505 in FIG. 9) is the same as the processing when a new topic is created, further explanations will be omitted.

[0238] Back in FIG. 12, if no reply request is detected in step S607, it is checked in step S608 whether a browsing end instruction (e.g., pressing the terminal on-hook key) is issued or not and if the browsing end instruction is issued, the processing is finished and, for example, the display data of the initial screen is sent. If no browsing end instruction is detected, the process goes back to step S603 and a detailed display request will be detected.

[0239] Change and Deletion Processing of Message

[0240] Then, message change and deletion processing will be explained using the display screen examples in FIGS. 13A to 13D and FIG. 14 showing the processing in the local server.

[0241] For example, if the message change/deletion link is selected from the initial screen (FIG. 13A), the mobile radio communication terminal 7 sends the subscriber number and a message change/deletion processing request to the local server 4. In response to this instruction, the local server 4 searches the message DB 57 using the subscriber number of the mobile radio communication terminal 7 and extracts the message registered by the relevant subscriber (step S701).

[0242] Then, the local sever 4 sends the title of the message to be displayed and message ID to the mobile radio communication terminal 7 (step S702). As a result, the message title is displayed on the screen of the mobile radio communication terminal 7 (FIG. 13B).

[0243] When a desired title is selected from the title display screen in FIG. 13B and the key assigned to "Change" or "Delete" is pressed (step S703), the mobile radio communication terminal 7 sends a change or deletion request and the ID of the message requested to be changed or deleted to the local server 4. The local server 4 that receives this searches the message DB 57 using the message ID and sends items necessary for a predetermined detailed display format, for example, message text, voice flag, transmission date/time, sender handle name, etc. to the mobile radio communication terminal 7 (step S704). As a result, the whole text of the message appears on the screen of the mobile radio communication terminal 7 (FIG. 13C or 13D).

[0244] In this embodiment, the title, transmission date/time and text are displayed on the whole text display screen shown in FIGS. 13C and 13D, and in the case where message voice recording is linked, a link to the voice data ("voice message") is displayed.

[0245] If the user selects the link to the voice data, which is not shown in the figure, from the whole text display screen of the message at the time of a change, the same voice recording processing as that in step S510 and S511 in the new message creation processing explained using FIG. 9 is performed and the voice DB 53 is overwritten with the voice data received from the mobile radio communication terminal 7.

[0246] Likewise, if the user selects the link to the voice data not shown in the figure from the whole message text display screen (FIG. 13D) at the time of deletion, the same voice replay processing as the replay processing (step S606) of the voice data in the browsing processing explained using FIG. 12 is performed and it is possible to audit voice data attached to the message by the mobile radio communication terminal 7.

[0247] Moreover, though not described in the figure, it is also possible to provide a link to newly add voice data on the changed whole message text display screen at the time of change. In this case, selecting the voice addition link will perform the same voice recording processing as that in step S510 and S511 in the new message creation processing explained using FIG. 9 and the local server newly registers the voice data received from the mobile radio communication terminal 7 in the voice DB 53.

[0248] Back in FIG. 14, in step S705, it is checked whether a deletion instruction is received from the whole message text display screen at the time of deletion and if the deletion instruction is received, the message being displayed is deleted from the message DB 57 (step S706)

[0249] Furthermore, in step S707, if the key assigned to "Send" is pressed from the whole message text display screen at the time of change and the message is sent (step S707), the message DB 57 is overwritten with the received message (step S708).

[0250] In step S709, it is checked whether a change/deletion end (e.g., pressing the terminal on-hook key) is instructed or not and if the change/deletion end is instructed, the processing is finished and the display data of the initial screen is sent. If no end instruction is detected, the process goes back to step S703 and a change/deletion processing request is detected.

[0251] Message Transfer Processing

[0252] Then, the message transfer processing carried out by the central processing server 1 will be explained by using the flow chart shown in FIG. 15.

[0253] As described above, it is possible to specify the location of the receiver about individual messages created. That is, in FIG. 1, by the mobile radio communication terminal 7 located in the service area of zone A creating a message with location specification corresponding to zone B

and/or zone C, it is possible to specify so that no message is supplied to the subscribers who exist (or enter in the future) in zone A and the message is supplied only to the subscribers who exist (or enter in the future) in zone B and/or zone C (of course, the message is not supplied to subscribers who exist in zone B and/or zone C but do not match other conditions set in the message).

[0254] As explained by using **FIG. 9**, upon receipt of a new message, the local server checks its location specification field and if any place other than its own location is specified, the message is transferred to the central processing server **1**.

[0255] Upon receipt of the message (step **S401**), the central processing server 1 checks the location specification field (step **S402**) and specifies the local server to which the message is to be transferred using the location specification contents and the local server DB **63** (**FIG. 3**) (step **S403**). Then, the message is transferred to the local server as the transfer destination (step **S404**). If there are a plurality of transfer destinations, the message is copied and transferred to each local server.

[0256] Each local server that has received the message from the central processing server **1** stores the message received in the message DB. As described before, if voice data is attached to the message, voice data is also attached to the message transferred from the local server to the central processing server **1**. When the message is transferred from the central processing server **1** to the local server as the transfer destination, the message with voice data attached will be transferred with voice data attached.

[0257] Each local server that has received the message with voice data attached stores the attached voice data in its own voice DB **53** and stores other character messages in the message DB **57**.

[0258] Broadcast Processing-Data Setting

[0259] Next, the broadcast processing will be explained. As described above, this system can supply information to the user by broadcasting. Broadcasting is a service of supplying display (or voice) data virtually continuously as in the case of radio, TV or character broadcasting.

[0260] In order to provide such a broadcast service, it is necessary to set the local server so that the local server can recognize data to be broadcast beforehand. Therefore, the setting of broadcast data will be explained using **FIGS. 16A** to **16D** first. In the following explanations, the case of setting at the mobile radio communication terminal **7** will be explained as an example, but as in the case of message processing, this setting can also be performed from devices connected to the Internet such as the information supply server **3**.

[0261] First, when a link to the broadcast setting screen provided on the initial screen (**FIG. 16A**) is selected and the mobile radio communication terminal **7** presses the key assigned to "Send", the broadcast setting screen is displayed (**FIG. 16B**). Selecting the transmission setting screen link ("Transmission setting") from the broadcast setting screen and pressing the key corresponding to "Send" will display the transmission setting screen (**FIG. 16C**).

[0262] As in case of the message creation screen (**FIG. 8B**), the transmission settings screen is provided with a receiver specification screen link, location specification screen link, date & time specification screen link and hobby specification screen link. Screen display and the setting contents on each screen when these links are selected are as described above using **FIGS. 8C** to **8F**, and therefore overlapping explanations will be omitted.

[0263] The transmission setting screen is further provided with a transmission data setting screen link and selecting this link and pressing the key corresponding to "Send" will display the transmission data setting screen (**FIG. 16D**).

[0264] The transmission data settings screen is provided with:

[0265] 1) Data source address input field

[0266] Field to specify supply source of broadcast data, for example IP address

[0267] 2) Download necessary/unnecessary setting field

[0268] Field to specify whether there is software to be downloaded to mobile radio communication terminal **7** separately in order for mobile radio communication terminal **7** to process broadcast data

[0269] 3) Download address input field

[0270] Field to input information to specify download software stored in local server, for example, URL, when "Necessary" is set in download necessary/unnecessary setting field

[0271] 4) Port number field

[0272] Field to enter port number to specify application program of mobile radio communication terminal **7** that processes broadcast data

[0273] 5) Terminal mandatory function specification field

[0274] Displaying fine data such as stock price at mobile radio communication terminal **7** requires dot matrix display on terminal side. Moreover, when voice data is broadcast, it is not possible to hear replayed voice if the speaker is not on the terminal side. Therefore, the function (equipment) necessary on the mobile radio communication terminal **7** side is specified to process data to be broadcast in this field.

[0275] When the key corresponding to "Set" on the transmission data setting screen is pressed, the setting contents of each setting screen linked from **FIG. 16C** and set value corresponding to each setting item included in **FIG. 16D** are sent to the local server. The setting contents is stored in the local server HDD **58**.

[0276] When the setting is performed, the local server accesses the address set in the source data address field and starts to receive broadcast data. Furthermore, the subscriber who performed broadcast setting transfers an application program necessary for FTP etc. to the address in the local server specified for the download address field.

[0277] Broadcast Data Processing

[0278] Then, the broadcast operation in this system will be explained using the screen display examples in **FIGS. 17A**

to 17C and the flow chart in **FIG. 18** to explain broadcast processing by the local server.

[0279] As explained using **FIG. 4**, when the mobile radio communication terminal **7** enters the service range of an AP which belongs to this system, a radio channel is established between the mobile radio communication terminal **7** and the AP and the subscriber number of the mobile radio communication terminal **7** is sent from the AP to the local server after the establishment of the radio channel.

[0280] Then, as explained in **FIG. 5**, the local server searches the user DB **8** using the subscriber number received from the AP and authenticates that the subscriber is a registered user. Then, when processing about the push message is completed, broadcast data processing is performed (steps S206 and S207).

[0281] More specifically, of the broadcast data preset at the local server, the display data of the channel selection screen **(FIG. 17A)** about the broadcast data that matches the subscriber setting of the mobile radio communication terminal **7** is sent (step S30) . This display data also includes the channel number information of each broadcast channel.

[0282] The channel setting screen shows broadcast channels that the user can receive and the individual radio buttons are placed next to the broadcast channels. When the user wants to receive one of these channels, the user selects the radio button corresponding to the one channel and presses the key corresponding to "Receive".

[0283] This channel selection is detected by the local server (step S302) and the local server reads the setting contents about the selected broadcast data from the HDD **58** and checks whether the hardware necessary for processing of the selected broadcast data is mounted in the user's terminal, that is, mobile radio communication terminal **7** or not (step S303). This check, for example, can be performed by the local server sending an inquiry command to the mobile radio communication terminal **7** and the mobile radio communication terminal **7** checking the spec of the terminal and sending back in response to this command.

[0284] As a result of the check, if the mobile radio communication terminal **7** does not have the dot matrix display even if the broadcast data requiring the dot matrix display has been selected, for example, a message that processing of the selected broadcast channel is not possible by the terminal is sent to the user (step S304) and the channel selection screen is returned to.

[0285] When the terminal **7** satisfies hardware-like requirements, it is then checked whether an application program necessary for processing of the broadcast data is required or not (step S305). This check references the contents of the broadcast data setting and checks whether the application program is necessary or not and also checks whether the mobile radio communication terminal **7** has the necessary application program or not.

[0286] Whether the mobile radio communication terminal **7** has the necessary application program or not can be checked, for example, by registering the application program downloaded in the past in the record of the user DB **8** for each user and referencing the user DB **8** in the processing in step S305.

[0287] In step S305, if it is decided that processing the selected broadcast data requires the mobile radio communication terminal **7** to download the software, the message notifying that and an inquiry screen **(FIG. 17B)** having the download link are sent (step S306).

[0288] When the user selects the download link ("Download") and presses the key corresponding to "Select", a download request is sent from the mobile radio communication terminal **7** to the local server **4** (step S307) and the application program is transferred from the local server **4** to the mobile radio communication terminal **7** according to a protocol such as HTTP and FTP (step S308). When the transfer of the application program is completed, the local server **4** registers the transferred application in the user DB. When the application program is transferred, the mobile radio communication terminal **7** starts to receive the broadcast data (step S309).

[0289] Furthermore, in step S305, if it is decided that the mobile radio communication terminal **7** can process broadcast data without downloading the software, inquiry screen data as to whether reception should be started or not is sent (step S310). When the mobile radio communication terminal **7** displays this data **(FIG. 17C)**, the user selects "Start reception" on the screen and presses the key corresponding to "Select", the mobile radio communication terminal **7** starts to receive broadcast data using the broadcast channel number received together with the display data of **(FIG. 17A)** (step S309).

[0290] As described above, the local server **4** always receives already set broadcast data and continues to send this broadcast data using a predetermined channel. Furthermore, the broadcast data is sent to the port number specified in the port number field of the broadcast data setting (port number assigned to the application program that performs broadcast data processing in the mobile radio communication terminal **7**).

[0291] Thus, the mobile radio communication terminal **7** performs processing such as display based on the broadcast data received by the broadcast data processing application program.

[0292] As explained above, since it is possible to specify various transmission destination conditions other than addresses for the created message, it is possible to supply information with the limited time and location, for example, sending a message so that only "subscribers who are in the premises of station A between AM 10:00 to 1:00 PM on September 8" can browse. Or since it is also possible to specify personal information such as a taste of a registered subscriber, it is also possible to specify, for example, "subscribers who are in the premises of station B and who are interested in sports", etc. In this way, viewed from the information supplier side, it is possible to supply information narrowing the range of the target and perform more effective advertisement when distributing advertisement, etc.

[0293] Furthermore, even with a message other than advertisement, if a message set so that "a message is pushed when a specific person comes to a specific place" is created, it is also possible to use the message in such a way that the contact address will be notified when the person who is late for an appointment arrives at the meeting place.

[0294] Moreover, when receiving information, specifying the information that one wants to receive (browse) beforehand makes it possible to acquire only the desired information, and therefore it is less likely to see unnecessary information and it is possible to acquire necessary information effectively.

[0295] As explained above, the information supply system according to the present invention allows both the party who supplies information and the party who receives the information to specify the receiver of the information and specify desired information, making it possible to acquire effective information supply and efficient information acquisition.

What is claimed is:

1. An information supply system that has at least one local server means having a service area with a predetermined range and a central server means for connecting between said local server means and supplies pre-stored information to a service subscriber terminal existing in said service area, comprising:

information database means for storing information with each of which a transmission destination specification condition is associated;

subscriber database means for storing an information reception condition set for each of said service subscriber terminals;

information selecting means for comparing said information reception condition corresponding to said service subscriber terminal existing in the service area of said local server means with said transmission destination specification condition using said information database means and said subscriber database means and selecting only information having said transmission destination specification condition that satisfies said information reception condition; and

information supplying means for presenting only the information selected by said information selecting means to said service subscriber terminal existing in said service area.

2. The information supply system according to claim 1, wherein said information database means, said information selecting means and said information supplying means are provided in each of said local server means.

3. The information supply system according to claim 1, wherein said transmission destination specification condition includes at least one of location specification conditions that specify the local server means as the transmission destination, period specification conditions that specify the period for supplying information and information reception condition specification conditions that specify at least a value of said information reception conditions.

4. The information supply system according to claim 1, wherein said local server means further comprising:

subscriber terminal detecting means for detecting that said service subscriber terminal has moved to the own service area; and

channel controlling means for establishing a radio channel with said service subscriber terminal that has moved and informing said information selecting means of the establishment of the radio channel.

5. The information supply system according to claim 1,

wherein a push condition to specify whether auto distribution is available or not is specified for each piece of information stored in said information database means, and

said information supplying means automatically transmits the information for which said push condition specifies that auto distribution is available from among the information pieces selected by said information selecting means to said service subscriber terminal existing in said service area.

6. The information supply system according to claim 1, wherein said information supplying means further supplies broadcast data virtually consecutively supplied to said service subscriber terminal as said information.

7. The information supply system according to claim 1, wherein said information supplying means presents said information in response to information request from said service subscriber terminal existing in said service area.

8. The information supply system according to claim 1, wherein said central server means distributes the relevant reception information to necessary local server means based on said transmission destination specification condition of the information received from each of said local server means.

9. A control method of information supply system which includes at least one local server means having a service area with a predetermined range and a central server means to connect between said local server means, information database means for storing information with which transmission destination specification conditions are associated and subscriber database means for storing information reception conditions set for each of service subscriber terminals and which supplies pre-stored information to said service subscriber terminal existing in said service area, comprising:

an information selecting step of comparing said information reception condition corresponding to said service subscriber terminal existing in the service area of said local server means with said information transmission destination specification condition using said information database means and said subscriber database means and selecting only information having said transmission destination specification condition that satisfies said information reception condition; and

an information supplying step of presenting only the information selected by said information selecting step to said service subscriber terminal existing in said service area.

10. The information supply system control method according to claim 9, wherein both said information selecting step and said information supplying step are executed by said local server means.

11. The information supply system control method according to claim 9, wherein said transmission destination specification condition includes at least one of location specification conditions for specifying the local server means as the transmission destination, period specification conditions for specifying the period of supplying information and information reception condition specification conditions for specifying at least some values of said information reception conditions.

12. The information supply system control method according to claim 9, further comprising:

a subscriber terminal detecting step, performed at said local server means, of detecting that said service subscriber terminal has moved to its own service area; and

a channel control step, performed at said local server means, of establishing a radio channel with said service subscriber terminal that has moved and notifying the information to said information selecting step.

13. The information supply system control method according to claim 9,

wherein a push condition for specifying whether auto distribution is available or not is specified for each piece of information stored in said information database means, and

said information supplying step automatically transmits the information for which said push condition specifies that auto distribution is available from among the information pieces selected by said information selecting step to said service subscriber terminal existing in said service area.

14. The information supply system control method according to claim 9, wherein said information supplying step provides broadcast data virtually continuously supplied to said service subscriber terminal as said information.

15. The information supply system control method according to claim 9, wherein said information supply step presents said information in response to information request from said service subscriber terminal existing in said service area.

16. The information supply system control method according to claim 9, wherein said central server means has a step of distributing relevant reception information to the necessary local server means based on said transmission destination specification condition of the information received from each of said local server means.

* * * * *

US 20020052796A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2002/0052796 A1

Tadokoro et al. (43) Pub. Date: May 2, 2002

(54) METHOD AND SYSTEM FOR PROVIDING SERVICES

(76) Inventors: Mitsuyoshi Tadokoro, Sagamihara (JP); Yuko Izumi, Isehara (JP)

Correspondence Address:
TOWNSEND AND TOWNSEND AND CREW, LLP
TWO EMBARCADERO CENTER
EIGHTH FLOOR
SAN FRANCISCO, CA 94111-3834 (US)

(21) Appl. No.: 09/888,259

(22) Filed: Jun. 21, 2001

(30) Foreign Application Priority Data

Oct. 27, 2000 (JP) P2000-329719

Publication Classification

(51) Int. Cl.[7] G06F 17/60

(52) U.S. Cl. 705/26

(57) ABSTRACT

The present invention provides techniques for allowing a user access to a plurality of services through a common access point. A service providing system includes the common access point and allows the user access with, for example, one user identifier (ID) and password. The user then requests a specific service and the service providing system, if it does not have the specific service, manages access to another service providing system to provide the specific service to the user. The services provided may be common or segregated by a criteria, such as area. For example a user in a certain area may receive services from service providers in his/her area.

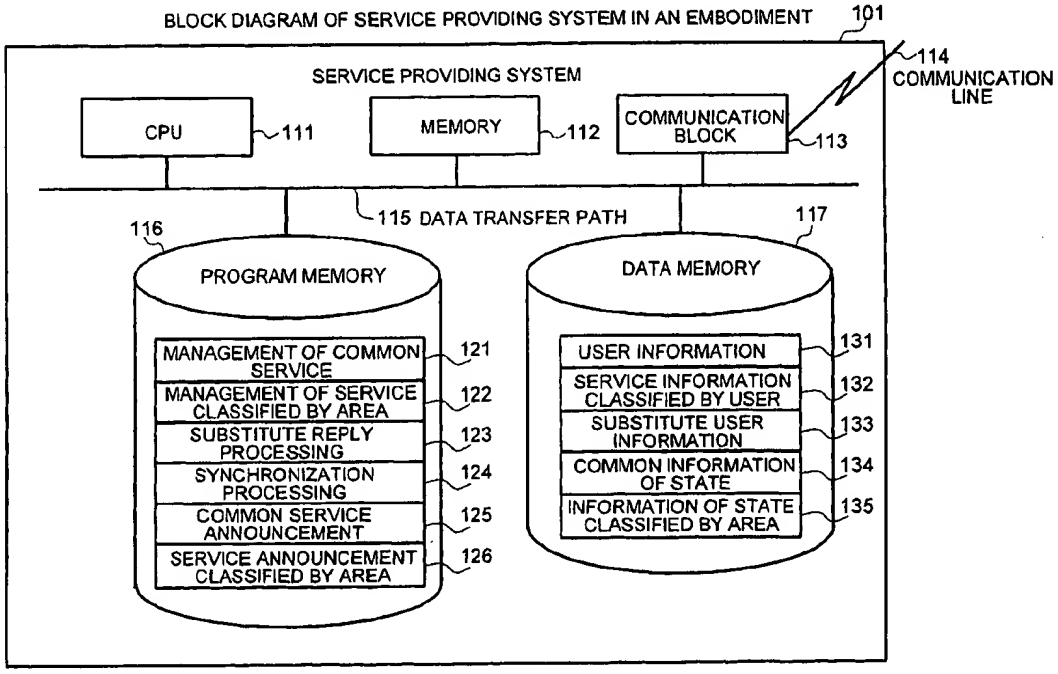


FIG.1

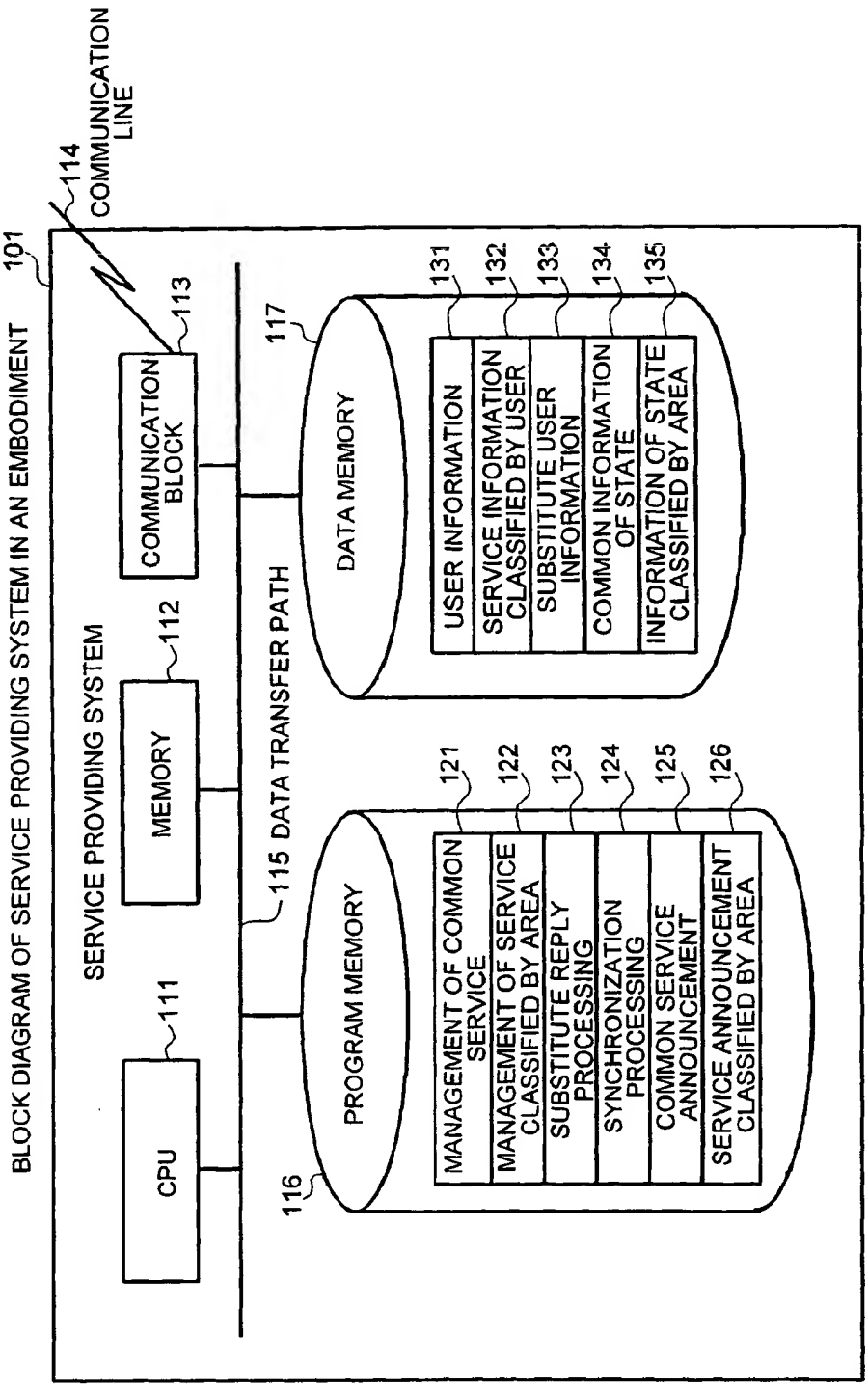

BLOCK DIAGRAM OF SERVICE PROVIDING SYSTEM IN AN EMBODIMENT
101
SERVICE PROVIDING SYSTEM
114
COMMUNICATION LINE
113
COMMUNICATION BLOCK
112
MEMORY
111
CPU
115 DATA TRANSFER PATH
117
DATA MEMORY
131
USER INFORMATION
132
SERVICE INFORMATION CLASSIFIED BY USER
133
SUBSTITUTE USER INFORMATION
134
COMMON INFORMATION OF STATE
135
INFORMATION OF STATE CLASSIFIED BY AREA
116
PROGRAM MEMORY
121
MANAGEMENT OF COMMON SERVICE
122
MANAGEMENT OF SERVICE CLASSIFIED BY AREA
123
SUBSTITUTE REPLY PROCESSING
124
SYNCHRONIZATION PROCESSING
125
COMMON SERVICE ANNOUNCEMENT
126
SERVICE ANNOUNCEMENT CLASSIFIED BY AREA

FIG.2

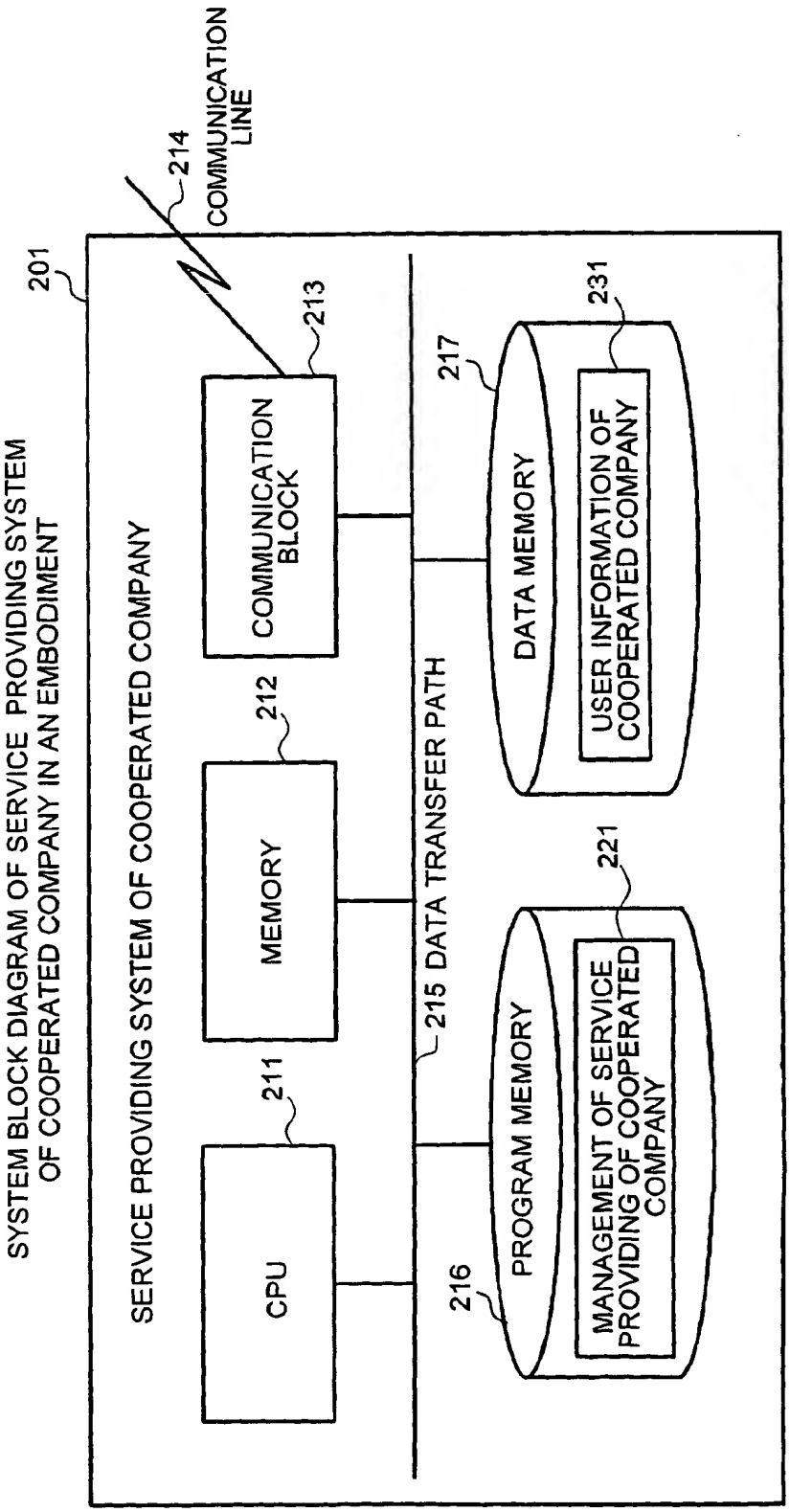

SYSTEM BLOCK DIAGRAM OF SERVICE PROVIDING SYSTEM OF COOPERATED COMPANY IN AN EMBODIMENT
201
SERVICE PROVIDING SYSTEM OF COOPERATED COMPANY
214
COMMUNICATION LINE
213
COMMUNICATION BLOCK
212
MEMORY
211
CPU
215 DATA TRANSFER PATH
217
DATA MEMORY
231
USER INFORMATION OF COOPERATED COMPANY
216
PROGRAM MEMORY
221
MANAGEMENT OF SERVICE PROVIDING OF COOPERATED COMPANY

FIG.3

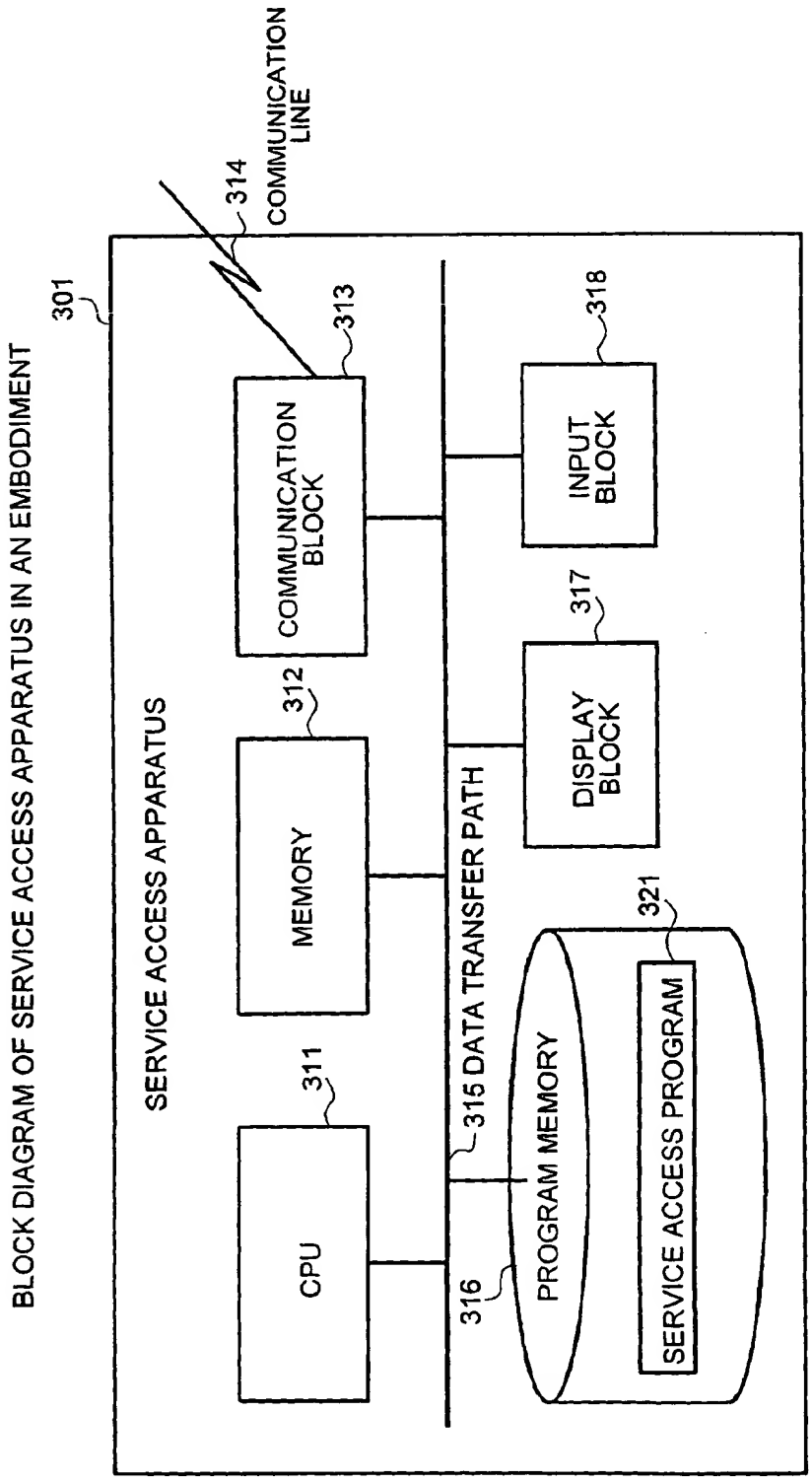

BLOCK DIAGRAM OF SERVICE ACCESS APPARATUS IN AN EMBODIMENT
301
SERVICE ACCESS APPARATUS
314
COMMUNICATION LINE
313
COMMUNICATION BLOCK
318
INPUT BLOCK
312
MEMORY
317
DISPLAY BLOCK
315 DATA TRANSFER PATH
321
PROGRAM MEMORY
SERVICE ACCESS PROGRAM
311
CPU
316

FIG.4

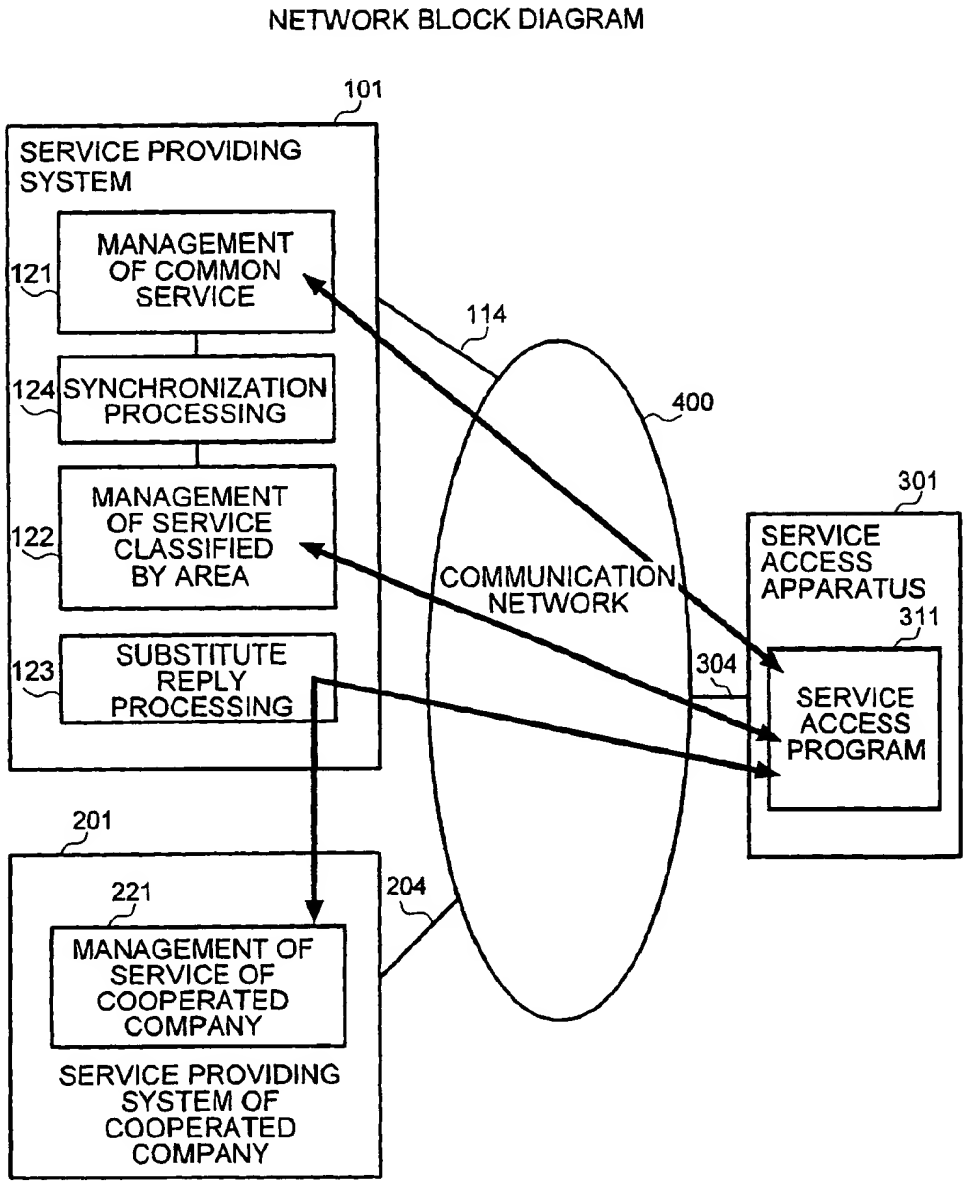

NETWORK BLOCK DIAGRAM
101
SERVICE PROVIDING SYSTEM
121
MANAGEMENT OF COMMON SERVICE
124
SYNCHRONIZATION PROCESSING
122
MANAGEMENT OF SERVICE CLASSIFIED BY AREA
123
SUBSTITUTE REPLY PROCESSING
114
400
COMMUNICATION NETWORK
301
SERVICE ACCESS APPARATUS
311
304
SERVICE ACCESS PROGRAM
201
221
MANAGEMENT OF SERVICE OF COOPERATED COMPANY
204
SERVICE PROVIDING SYSTEM OF COOPERATED COMPANY

FIG.4-1

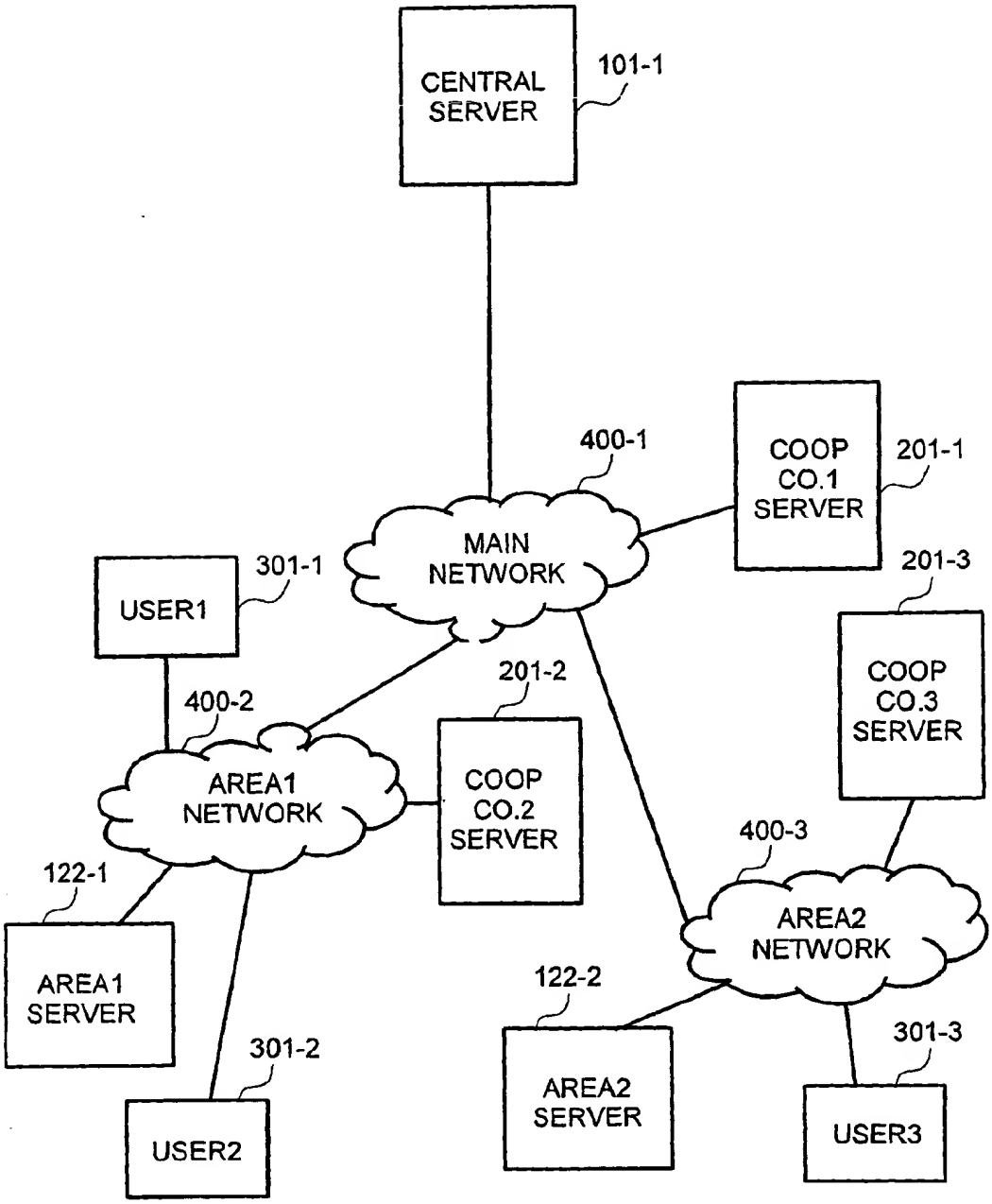

CENTRAL SERVER
101-1
MAIN NETWORK
400-1
COOP CO.1 SERVER
201-1
USER1
301-1
COOP CO.3 SERVER
201-3
400-2
201-2
AREA1 NETWORK
COOP CO.2 SERVER
400-3
122-1
AREA2 NETWORK
AREA1 SERVER
122-2
301-3
301-2
AREA2 SERVER
USER3
USER2

FIG.5

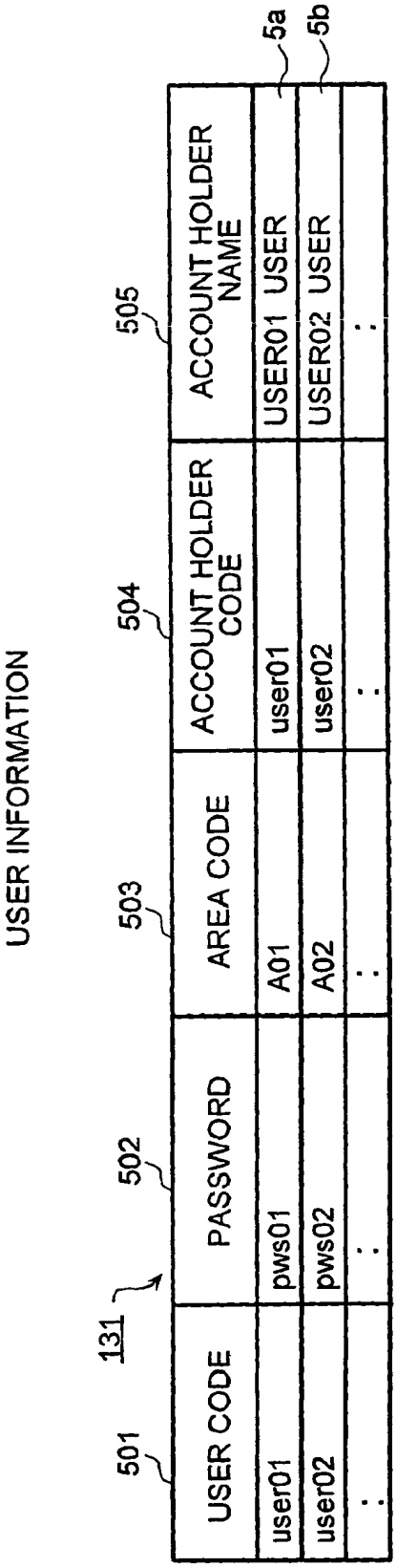

USER INFORMATION
131
501
502
503
504
505
5a
5b
USER CODE
PASSWORD
AREA CODE
ACCOUNT HOLDER CODE
ACCOUNT HOLDER NAME
user01
pws01
A01
user01
USER01 USER
user02
pws02
A02
user02
USER02 USER

FIG.6

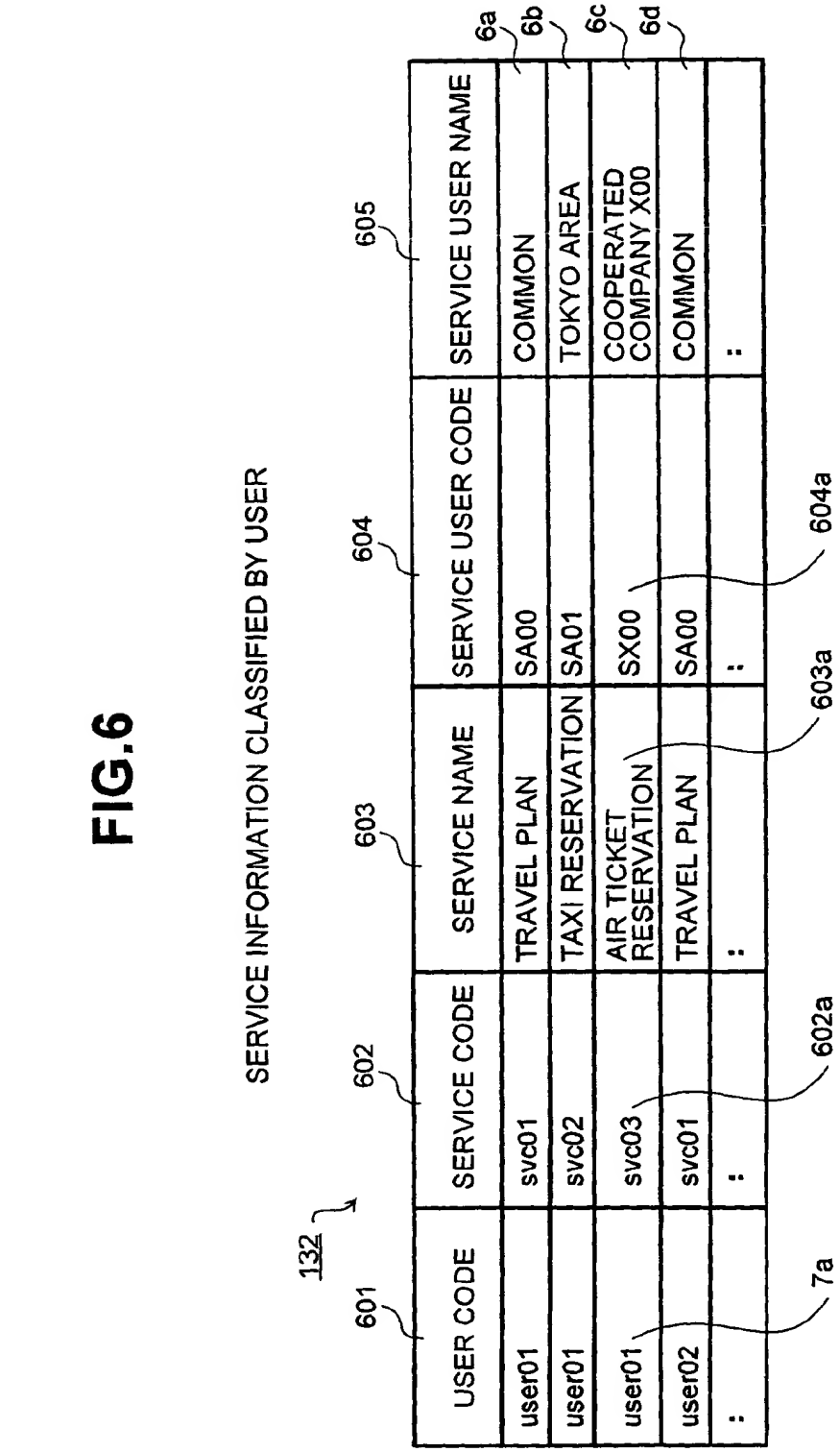

SERVICE INFORMATION CLASSIFIED BY USER
132
601
602
603
604
605
6a
6b
6c
6d
602a
603a
604a
7a
USER CODE
SERVICE CODE
SERVICE NAME
SERVICE USER CODE
SERVICE USER NAME
user01
svc01
TRAVEL PLAN
SA00
COMMON
user01
svc02
TAXI RESERVATION
SA01
TOKYO AREA
user01
svc03
AIR TICKET RESERVATION
SX00
COOPERATED COMPANY X00
user02
svc01
TRAVEL PLAN
SA00
COMMON

FIG.7

SUBSTITUTE USER INFORMATION

133

701 SUBSTITUTE USER CODE	702 SUBSTITUTE PASSWORD	703 COOPERATED COMPANY CODE	704 USER CODE
Xuser01	Xpws01	X00	user01
:	:	:	:

701a 702a 703a 704a

FIG.8

COMMON INFORMATION OF STATE

134

801 USER CODE	802 SERVICE CODE	803 COOPERATED COMPANY USER CODE	804 STATE CODE	805 SERVICE PROVIDING TIME	
user01	svc01		login	2000/10/10 18:00 00	8a
user01	svc01		start	2000/10/10 18:01 00	8b
user01	svc01		end	2000/10/10 18:02 00	8c
user01	svc02		start	2000/10/10 18:03 00	8d
user01	svc02		end	2000/10/10 18:04 00	8e
user01	svc03	Xuser01	start	2000/10/10 18:05 00	8f
user01	svc03	Xuser01	end	2000/10/10 18:06 00	8g
user01	svc03	Xuser01	logout	2000/10/10 18:07 00	8h
user02	svc01		login	2000/10/10 18:10 00	8i
:	:	:	:	:	

801a 802a 803a

FIG.9

INFORMATION OF STATE CLASSIFIED BY AREA

135

901 AREA CODE	902 USER CODE	903 SERVICE CODE	904 STATE CODE	905 SERVICE PROVIDING TIME	
A01	user01	svc01	login	2000/10/10 18:00 00	9a
A01	user01	svc01	start	2000/10/10 18:01 00	9b
A01	user01	svc01	end	2000/10/10 18:02 00	9c
A01	user01	svc02	start	2000/10/10 18:03 00	9d
A01	user01	svc02	end	2000/10/10 18:04 00	9e
A01	user01	svc03	start	2000/10/10 18:05 00	9f
A01	user01	svc03	end	2000/10/10 18:06 00	9g
A01	user01	svc03	logout	2000/10/10 18:07 00	9h
:	:	:	:	:	

FIG.10

COOPERATED COMPANY USER INFORMATION

231

1001 USER CODE OF COOPERATED COMPANY	1002 PASSWORD OF COOPERATED COMPANY	1003 ACCOUNT HOLDER CODE OF COOPERATED COMPANY	1004 ACCOUNT HOLDER NAME OF COOPERATED COMPANY
Xuser01	Xpws01	Xuser01	A00 Ltd
Xuser11	Xpws11	Xuser11	XUSER01 XUSER
:	:	:	:

10a
10b

FIG.11

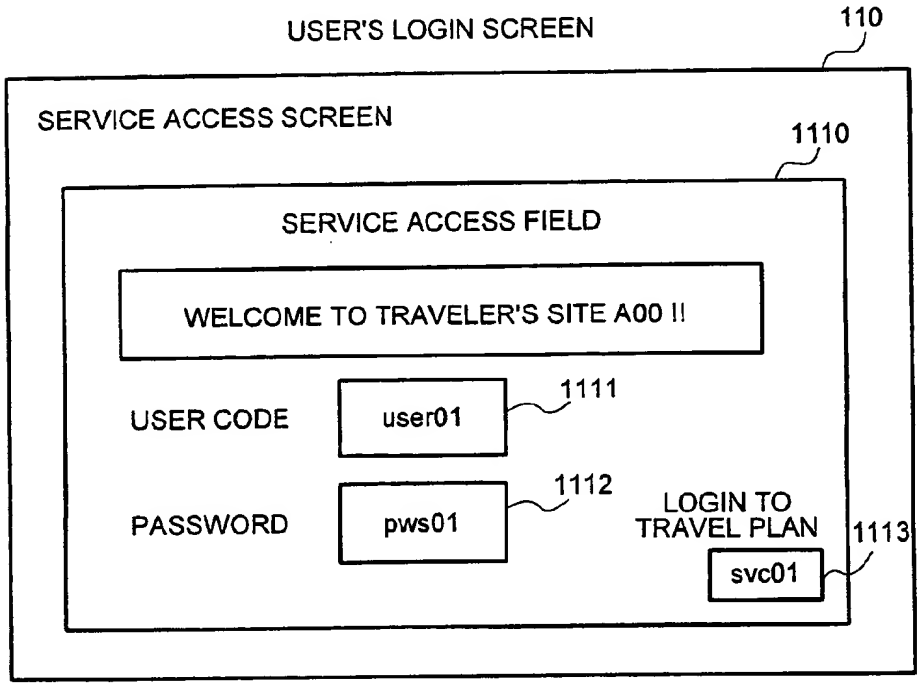


FIG.12

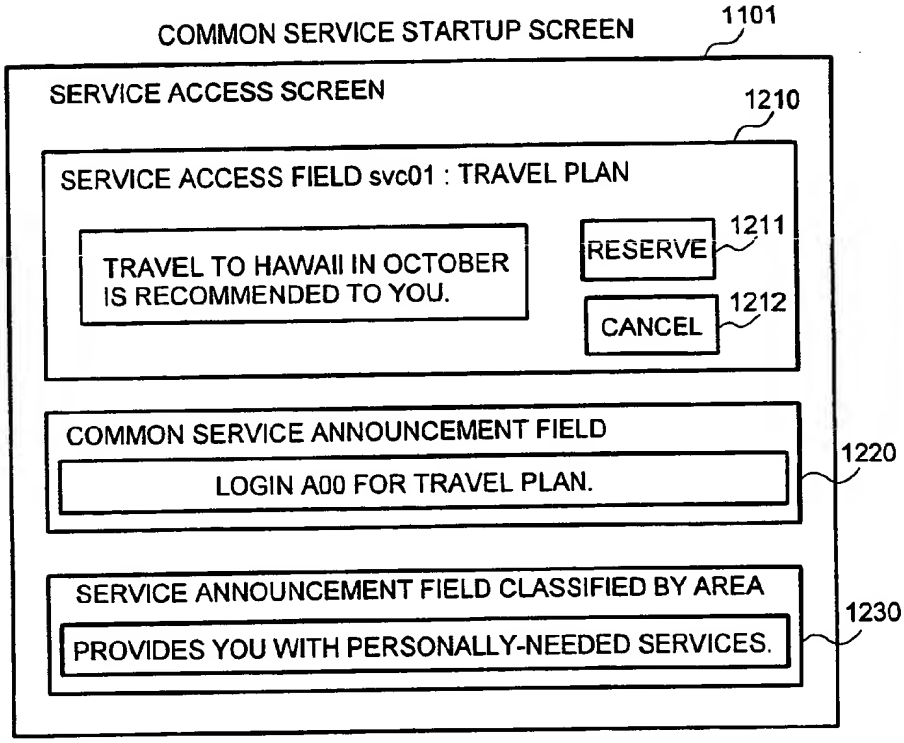

COMMON SERVICE STARTUP SCREEN
1101
SERVICE ACCESS SCREEN
1210
SERVICE ACCESS FIELD svc01 : TRAVEL PLAN
TRAVEL TO HAWAII IN OCTOBER IS RECOMMENDED TO YOU.
RESERVE
1211
CANCEL
1212
COMMON SERVICE ANNOUNCEMENT FIELD
LOGIN A00 FOR TRAVEL PLAN.
1220
SERVICE ANNOUNCEMENT FIELD CLASSIFIED BY AREA
PROVIDES YOU WITH PERSONALLY-NEEDED SERVICES.
1230

FIG.13

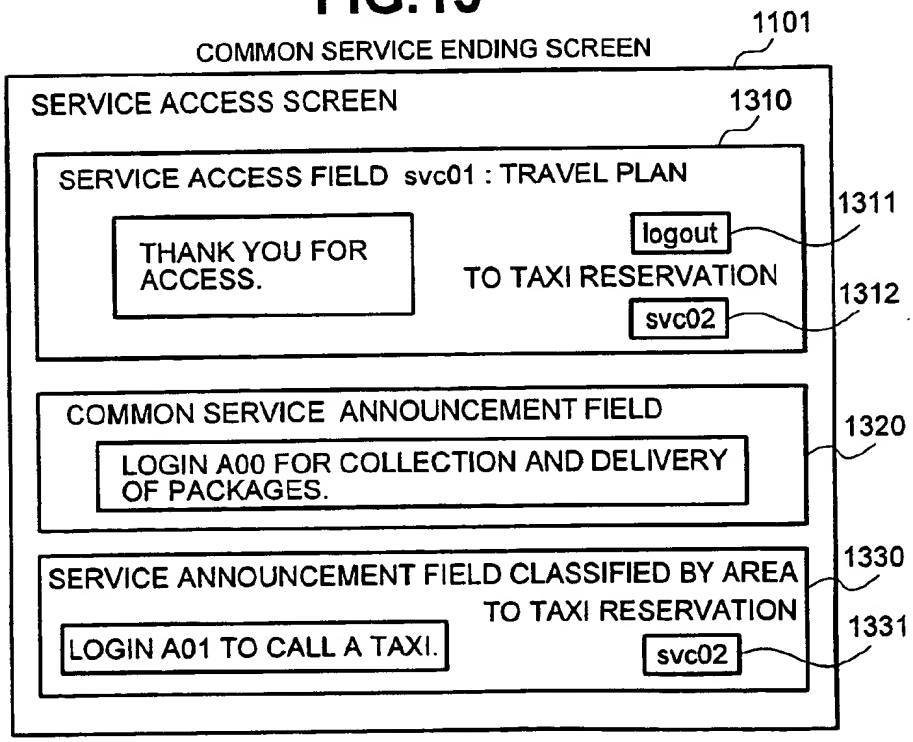

COMMON SERVICE ENDING SCREEN
1101
SERVICE ACCESS SCREEN
1310
SERVICE ACCESS FIELD svc01 : TRAVEL PLAN
THANK YOU FOR ACCESS.
logout
1311
TO TAXI RESERVATION
svc02
1312
COMMON SERVICE ANNOUNCEMENT FIELD
LOGIN A00 FOR COLLECTION AND DELIVERY OF PACKAGES.
1320
SERVICE ANNOUNCEMENT FIELD CLASSIFIED BY AREA
1330
TO TAXI RESERVATION
LOGIN A01 TO CALL A TAXI.
svc02
1331

FIG.14

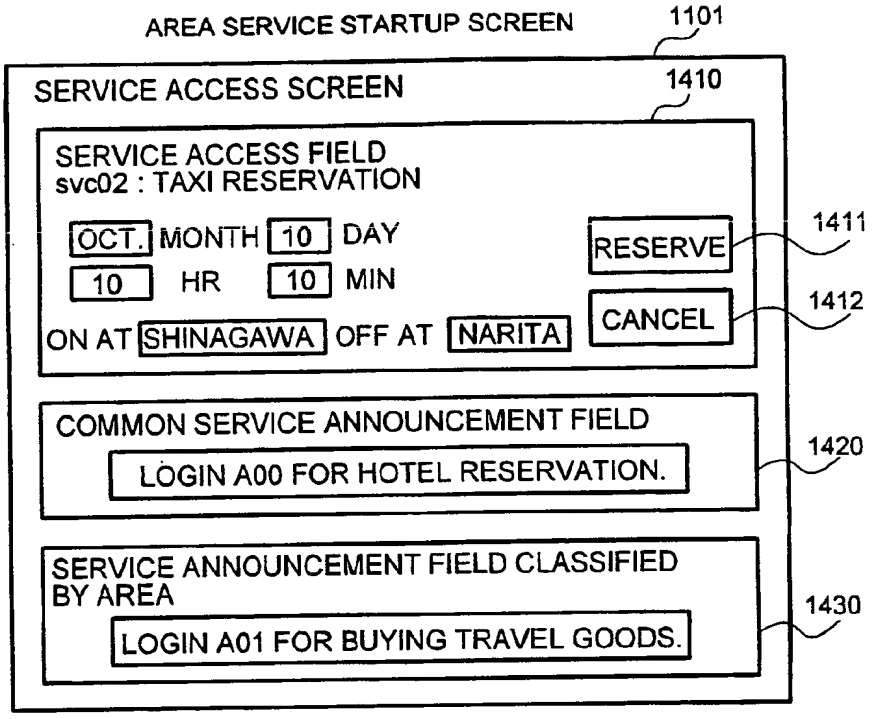

AREA SERVICE STARTUP SCREEN
1101
SERVICE ACCESS SCREEN
1410
SERVICE ACCESS FIELD
svc02 : TAXI RESERVATION
OCT. MONTH 10 DAY
10 HR 10 MIN
ON AT SHINAGAWA OFF AT NARITA
RESERVE
1411
CANCEL
1412
COMMON SERVICE ANNOUNCEMENT FIELD
LOGIN A00 FOR HOTEL RESERVATION.
1420
SERVICE ANNOUNCEMENT FIELD CLASSIFIED BY AREA
LOGIN A01 FOR BUYING TRAVEL GOODS.
1430

FIG.15

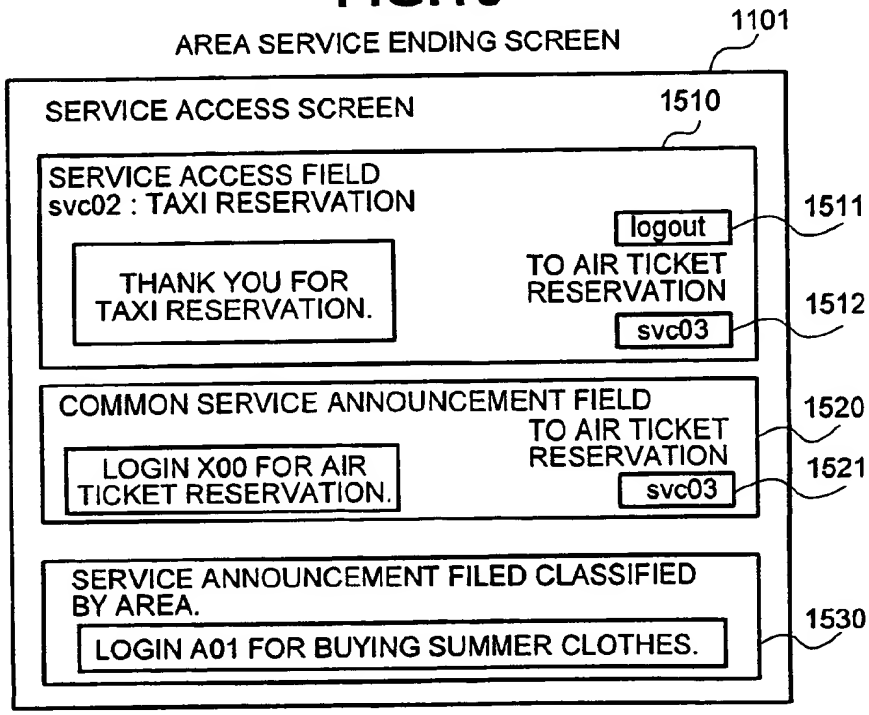

AREA SERVICE ENDING SCREEN
1101
SERVICE ACCESS SCREEN
1510
SERVICE ACCESS FIELD
svc02 : TAXI RESERVATION
THANK YOU FOR TAXI RESERVATION.
logout
1511
TO AIR TICKET RESERVATION
svc03
1512
COMMON SERVICE ANNOUNCEMENT FIELD
LOGIN X00 FOR AIR TICKET RESERVATION.
TO AIR TICKET RESERVATION
svc03
1520
1521
SERVICE ANNOUNCEMENT FILED CLASSIFIED BY AREA.
LOGIN A01 FOR BUYING SUMMER CLOTHES.
1530

FIG.16

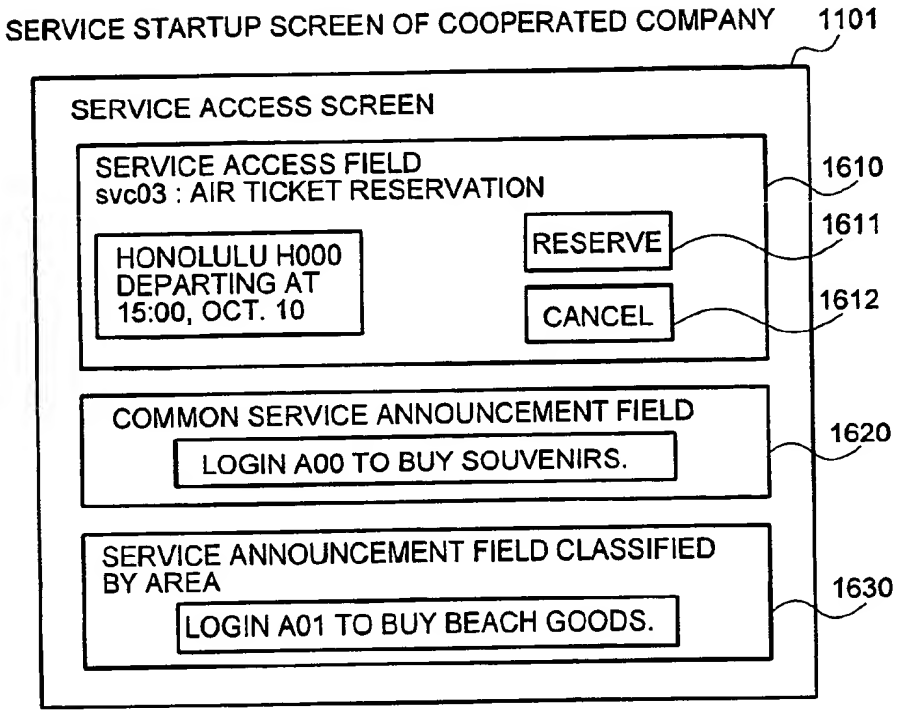

SERVICE STARTUP SCREEN OF COOPERATED COMPANY
1101
SERVICE ACCESS SCREEN
SERVICE ACCESS FIELD
svc03 : AIR TICKET RESERVATION
1610
HONOLULU H000
DEPARTING AT
15:00, OCT. 10
RESERVE
1611
CANCEL
1612
COMMON SERVICE ANNOUNCEMENT FIELD
LOGIN A00 TO BUY SOUVENIRS.
1620
SERVICE ANNOUNCEMENT FIELD CLASSIFIED BY AREA
LOGIN A01 TO BUY BEACH GOODS.
1630

FIG.17

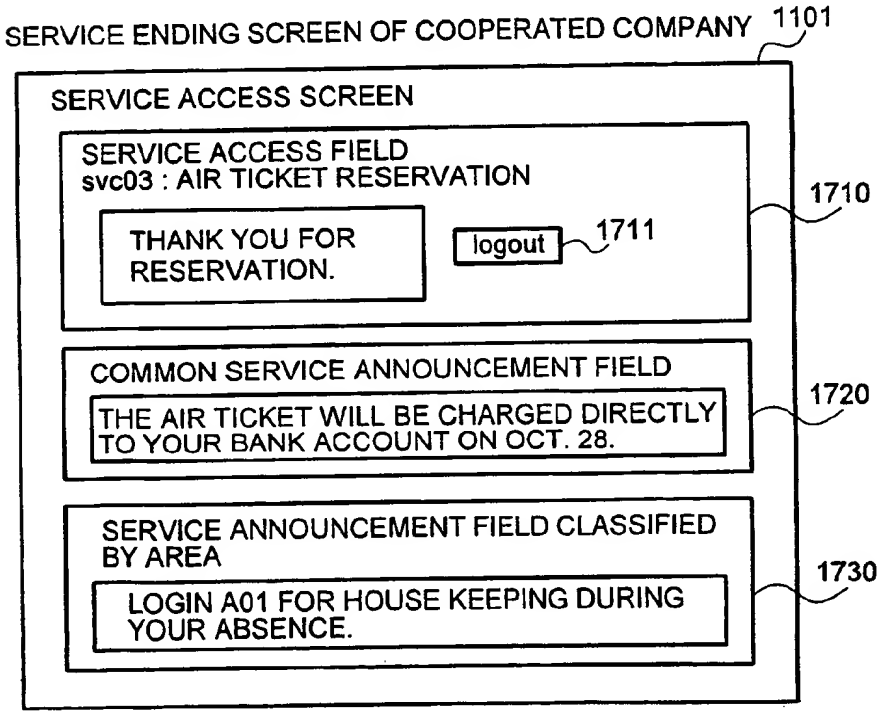

SERVICE ENDING SCREEN OF COOPERATED COMPANY
1101
SERVICE ACCESS SCREEN
SERVICE ACCESS FIELD
svc03 : AIR TICKET RESERVATION
1710
THANK YOU FOR
RESERVATION.
logout
1711
COMMON SERVICE ANNOUNCEMENT FIELD
THE AIR TICKET WILL BE CHARGED DIRECTLY TO YOUR BANK ACCOUNT ON OCT. 28.
1720
SERVICE ANNOUNCEMENT FIELD CLASSIFIED BY AREA
LOGIN A01 FOR HOUSE KEEPING DURING YOUR ABSENCE.
1730

FIG.18

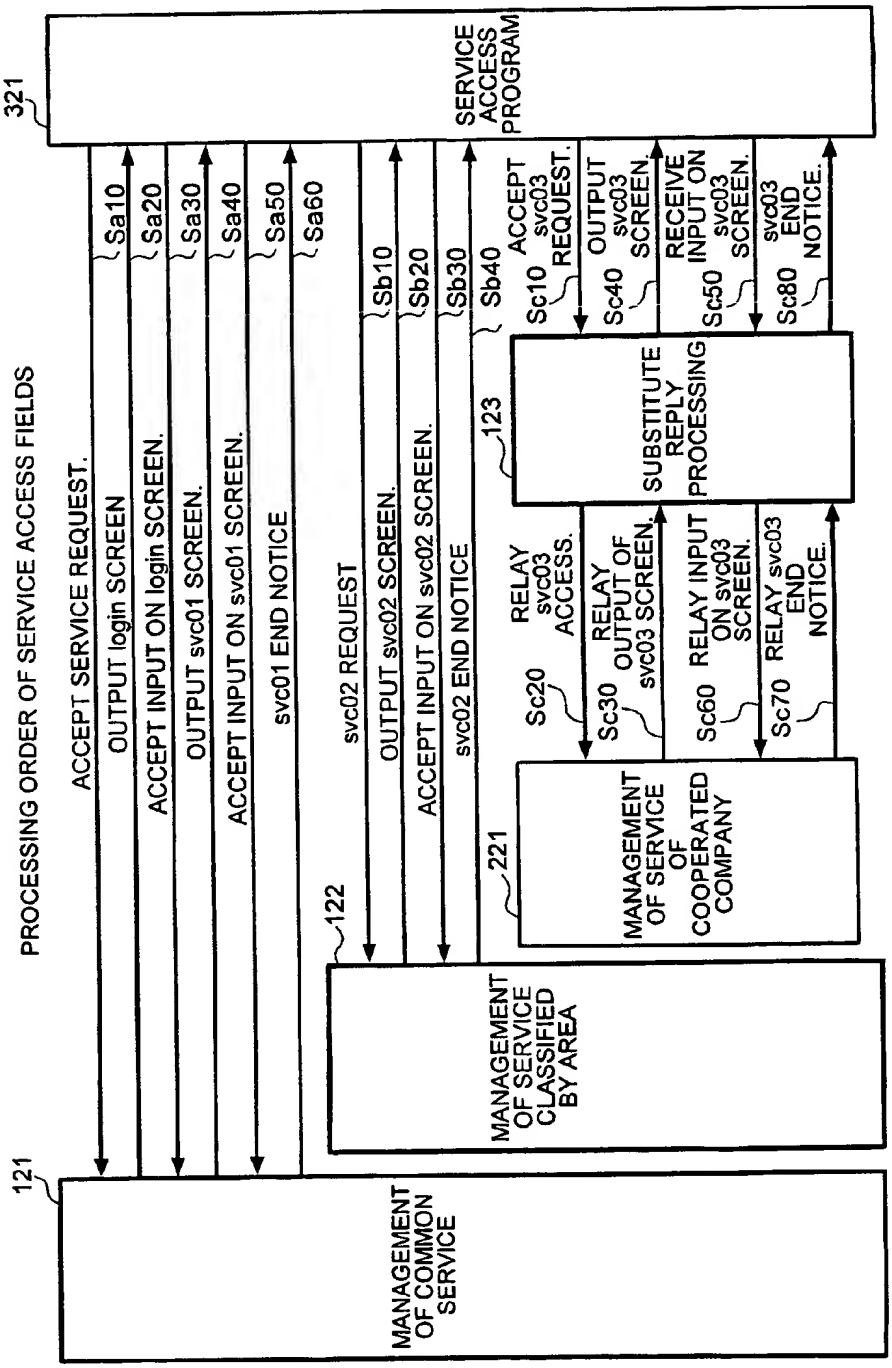

PROCESSING ORDER OF SERVICE ACCESS FIELDS
321
SERVICE ACCESS PROGRAM
ACCEPT SERVICE REQUEST.
OUTPUT login SCREEN
ACCEPT INPUT ON login SCREEN.
OUTPUT svc01 SCREEN.
ACCEPT INPUT ON svc01 SCREEN.
svc01 END NOTICE
svc02 REQUEST
OUTPUT svc02 SCREEN.
ACCEPT INPUT ON svc02 SCREEN.
svc02 END NOTICE
Sa10
Sa20
Sa30
Sa40
Sa50
Sa60
Sb10
Sb20
Sb30
Sb40
ACCEPT svc03 REQUEST.
OUTPUT svc03 SCREEN.
RECEIVE INPUT ON svc03 SCREEN.
svc03 END NOTICE.
Sc10
Sc40
Sc50
Sc80
123
SUBSTITUTE REPLY PROCESSING
RELAY svc03 ACCESS.
RELAY OUTPUT OF svc03 SCREEN.
RELAY INPUT ON svc03 SCREEN.
RELAY svc03 END NOTICE.
Sc20
Sc30
Sc60
Sc70
221
MANAGEMENT OF SERVICE OF COOPERATED COMPANY
122
MANAGEMENT OF SERVICE CLASSIFIED BY AREA
121
MANAGEMENT OF COMMON SERVICE

FIG.19

FLOWCHART OF MANAGEMENT OF COMMON SERVICE PROVIDING

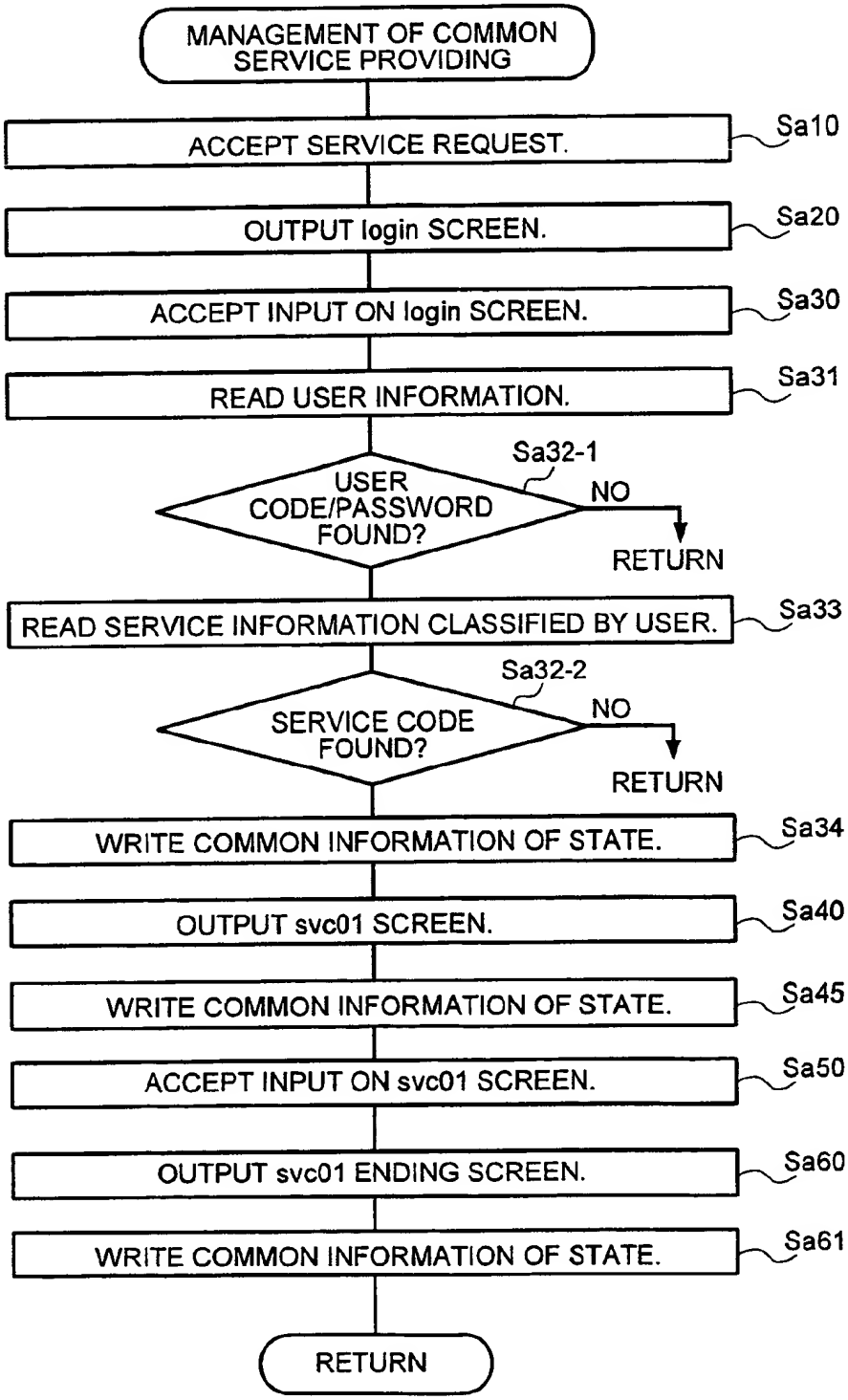


FIG.20

FLOWCHART OF MANAGEMENT OF SERVICE PROVIDING CLASSIFIED BY AREA

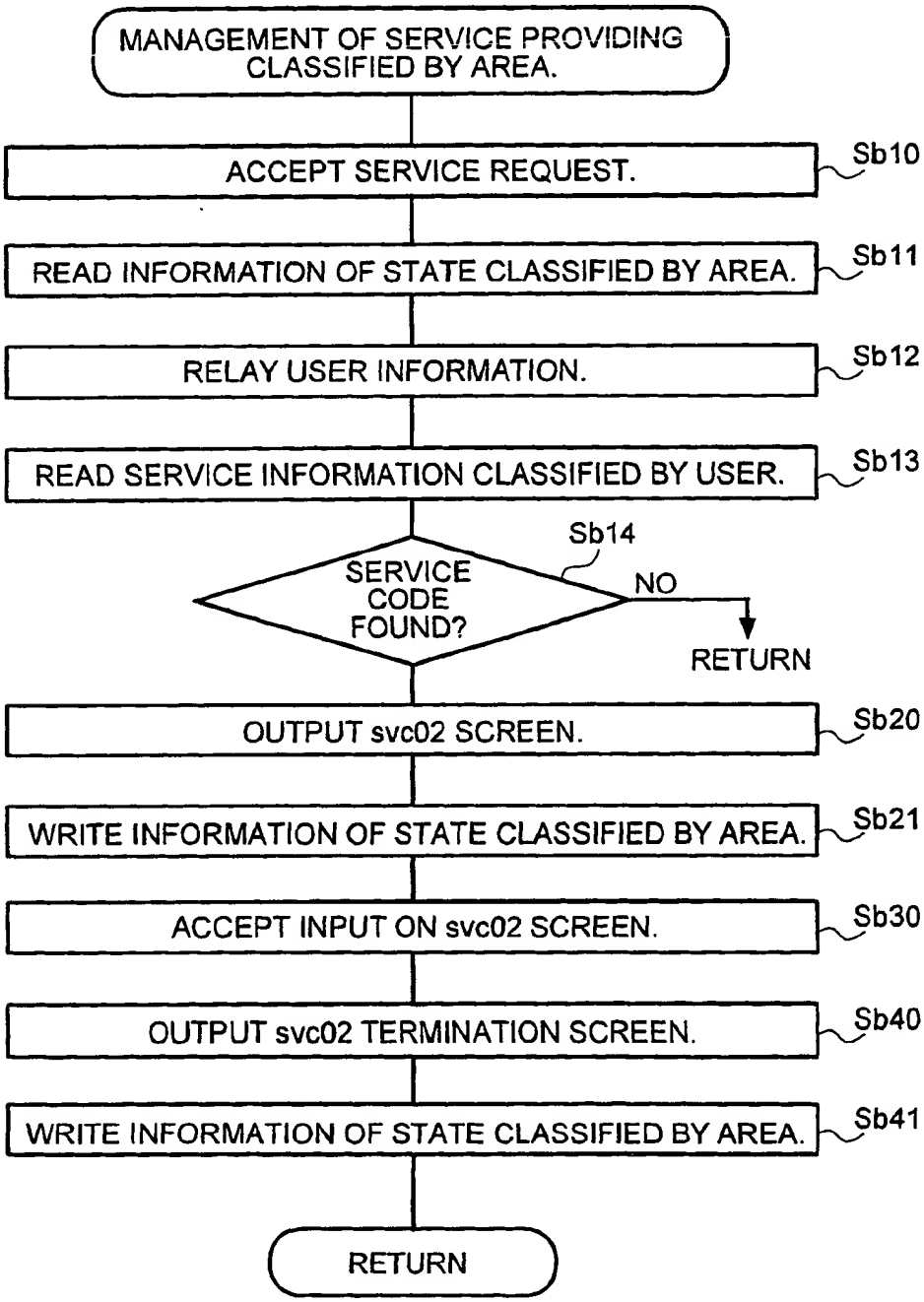


FIG.21

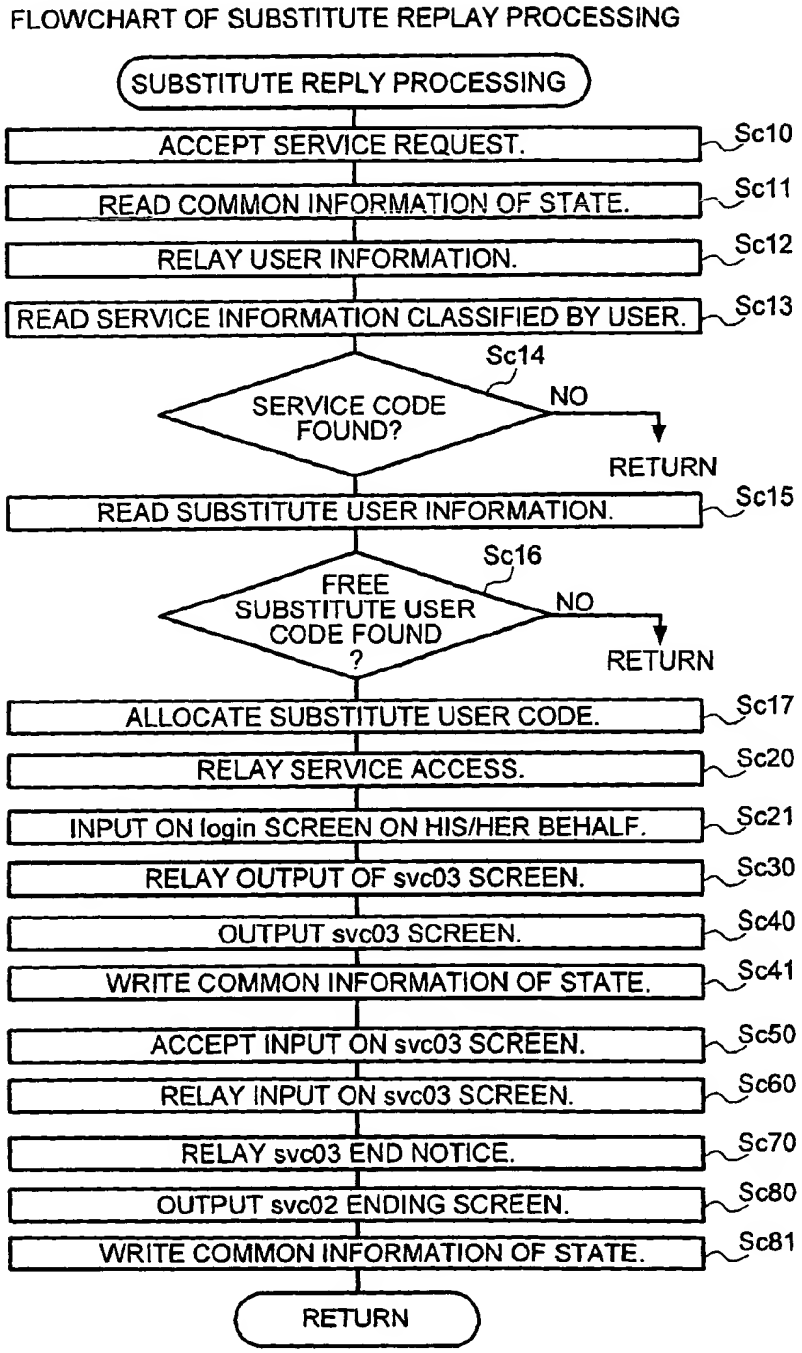

FLOWCHART OF SUBSTITUTE REPLAY PROCESSING
SUBSTITUTE REPLY PROCESSING
ACCEPT SERVICE REQUEST.
Sc10
READ COMMON INFORMATION OF STATE.
Sc11
RELAY USER INFORMATION.
Sc12
READ SERVICE INFORMATION CLASSIFIED BY USER.
Sc13
Sc14
SERVICE CODE FOUND?
NO
RETURN
READ SUBSTITUTE USER INFORMATION.
Sc15
Sc16
FREE SUBSTITUTE USER CODE FOUND ?
NO
RETURN
ALLOCATE SUBSTITUTE USER CODE.
Sc17
RELAY SERVICE ACCESS.
Sc20
INPUT ON login SCREEN ON HIS/HER BEHALF.
Sc21
RELAY OUTPUT OF svc03 SCREEN.
Sc30
OUTPUT svc03 SCREEN.
Sc40
WRITE COMMON INFORMATION OF STATE.
Sc41
ACCEPT INPUT ON svc03 SCREEN.
Sc50
RELAY INPUT ON svc03 SCREEN.
Sc60
RELAY svc03 END NOTICE.
Sc70
OUTPUT svc02 ENDING SCREEN.
Sc80
WRITE COMMON INFORMATION OF STATE.
Sc81
RETURN

FIG.22

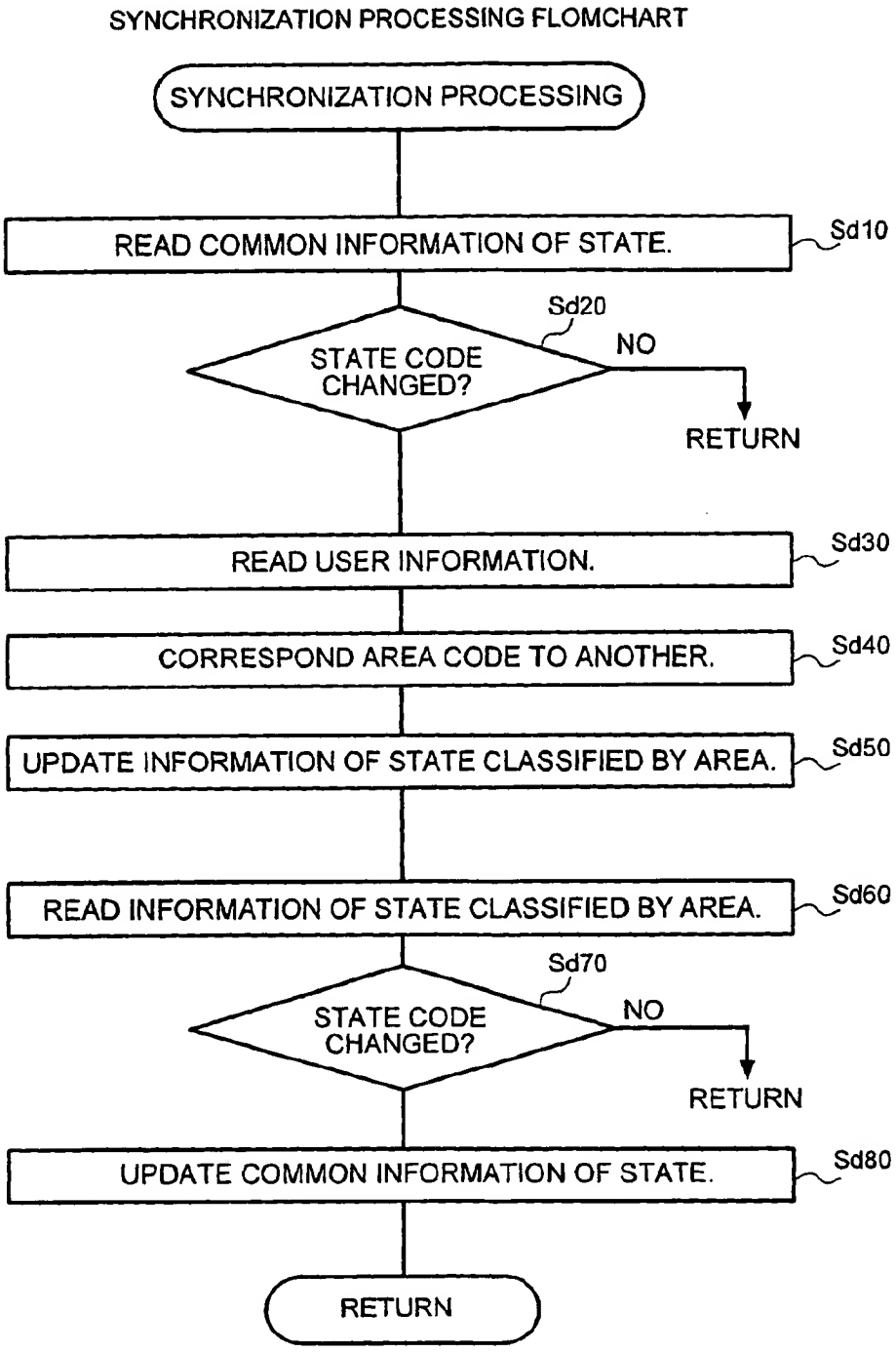

SYNCHRONIZATION PROCESSING FLOMCHART
SYNCHRONIZATION PROCESSING
READ COMMON INFORMATION OF STATE.
Sd10
Sd20
STATE CODE CHANGED?
NO
RETURN
READ USER INFORMATION.
Sd30
CORRESPOND AREA CODE TO ANOTHER.
Sd40
UPDATE INFORMATION OF STATE CLASSIFIED BY AREA.
Sd50
READ INFORMATION OF STATE CLASSIFIED BY AREA.
Sd60
Sd70
STATE CODE CHANGED?
NO
RETURN
UPDATE COMMON INFORMATION OF STATE.
Sd80
RETURN

FIG.23

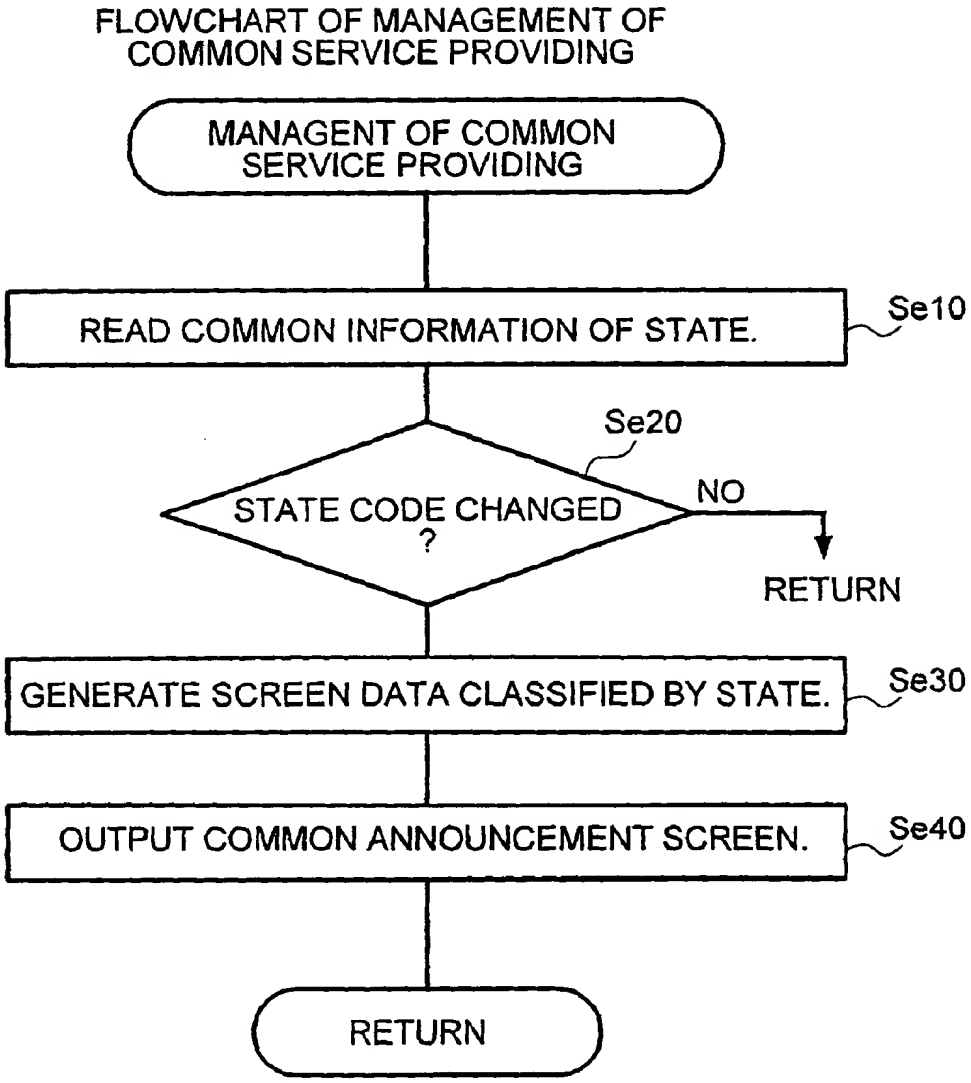

FLOWCHART OF MANAGEMENT OF COMMON SERVICE PROVIDING
MANAGENT OF COMMON SERVICE PROVIDING
READ COMMON INFORMATION OF STATE.
Se10
Se20
STATE CODE CHANGED ?
NO
RETURN
GENERATE SCREEN DATA CLASSIFIED BY STATE.
Se30
OUTPUT COMMON ANNOUNCEMENT SCREEN.
Se40
RETURN

FIG.24

FLOWCHART OF MANAGEMENT OF SERVICE ANNOUNCEMENT CLASSIFIED BY AREA

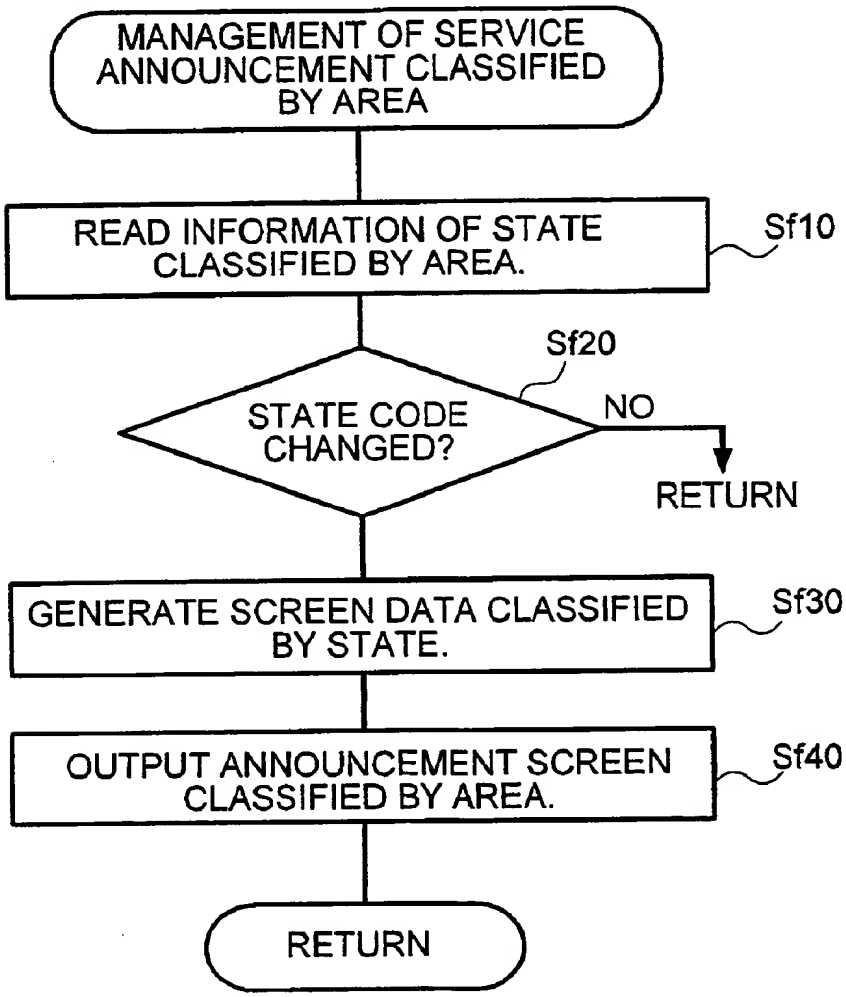
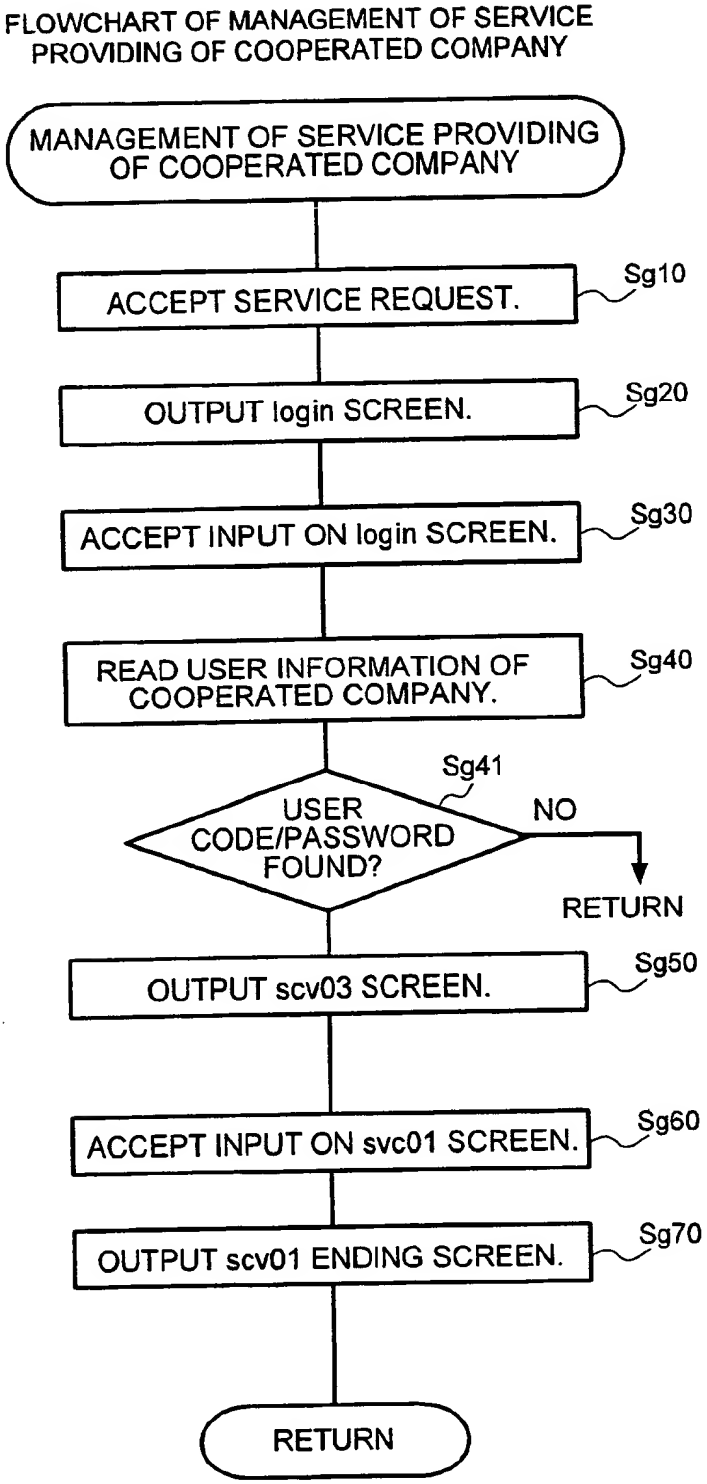


FIG.25


FLOWCHART OF MANAGEMENT OF SERVICE PROVIDING OF COOPERATED COMPANY
MANAGEMENT OF SERVICE PROVIDING OF COOPERATED COMPANY
ACCEPT SERVICE REQUEST.
Sg10
OUTPUT login SCREEN.
Sg20
ACCEPT INPUT ON login SCREEN.
Sg30
READ USER INFORMATION OF COOPERATED COMPANY.
Sg40
Sg41
USER CODE/PASSWORD FOUND?
NO
RETURN
OUTPUT scv03 SCREEN.
Sg50
ACCEPT INPUT ON svc01 SCREEN.
Sg60
OUTPUT scv01 ENDING SCREEN.
Sg70
RETURN

METHOD AND SYSTEM FOR PROVIDING SERVICES

CROSS-REFERENCES TO RELATED APPLICATIONS

[0001] This application is related to and claims priority from Japanese Patent Application No. 2000-329719, filed on Oct. 27, 2000.

BACKGROUND OF THE INVENTION

[0002] The present invention relates generally to electronic commerce over a communications network, for example, the Internet, and more particularly to providing a user a common access point to a plurality of services, some of which are segregated by location.

[0003] In recent years, there have appeared many sites that provide various services through their Internet WWW (World Wide Web) servers. Each user accesses those WWW servers via a WWW client so as to receive the services. To access a WWW server that provides commercial services, each user is requested to make a contract with a service company that runs the subject WWW server so as to be given a user ID and a password before he/she can uses the service.

[0004] A service provider of these services provided on the WWW, when permitting a user to use a service, often requests the user to send his/her personal information. This is to monitor whether or not the provided service is used in accordance with the contracted access condition and/or whether or not the service is used legally. Where there is a charge for the service, the provided service is monitored so as to calculate how much the user should be charged. Sometimes, the state of the user who is accessing the WWW server is monitored so as to use the results of the monitoring for market research. In these cases, each user, when he/she wants to access a service of a server, has been required to send his/her personal information to the server.

[0005] The conventional technique is burdensome to a user, since the user must register himself/herself in each server before using a service provided therefrom. The user is also forced to manage a pair of user ID and password for each time he/she registers to use a service. Thus, the more the user increases the number of services he/she uses, the more the user is forced to remember and manage his/her user IDs and passwords. Also the passwords may consist of incoherent alphanumerics further increasing the memory burden on the user. Consequently, the more the WWW comes to provide various useful services, the more the users' inconvenience is increased. In addition, users have come to feel uneasy about the diversion of their personal information, since their personal information is now held in servers around the world. These problems have been a bottle-neck for promoting information service markets on the WWW.

[0006] This bottleneck has adversely affected service providers, because the deepest interest of service providers is how to obtain more users, since the service providers make profits only from charges from their service users.

[0007] Thus there is a need to reduce the burden on the user in accessing services and hence increasing the opportunity of service providers in obtaining more users.

SUMMARY OF THE INVENTION

[0008] The present invention provides techniques for allowing a user access to a plurality of services through a common access point. A service providing system includes the common access point and allows the user access with, for example, one user identifier (ID) and password. The user then requests a specific service and the service providing system, if it does not have the specific service, manages access to another service providing system to provide the specific service to the user. The services provided may be common or segregated by a criteria, such as area. For example a user in a certain area may receive services from service providers in his/her area. In one embodiment the secondary service providing company sees only the common access service providing company as its user. Thus, the actual user is shielded from the secondary service providing company by the common access service providing company.

[0009] One embodiment of the present invention provides a method for a user to use a plurality of services through a common access point. The method includes, comprising: providing the user access to a first computer system. Next, responsive to a user request for a service of, the first computer system accesses a second computer system, having the service; and then the service is provided to said user from the second computer system via the first computer system.

[0010] A second embodiment of the present invention provides a service providing system for providing a service located on a computer system to a user. The user need only access the service providing system to obtain the service. The computer system is coupled to the service providing system via a communications network, for example, the Internet or an intra-net. The service providing system includes: a storage system; an access table stored in a first part of said storage system, having a user identification for access to the service providing system by the user; a mapping table stored in a second part of the storage system, having the user identification and associated login information, where the associated login information is for access by the service providing system to the service; and responsive to a request by the user for the service, software stored in a third part of the storage system for accessing the computer system using said login information and for obtaining the service for the user. The storage system may be one memory or a plurality of memories of volatile or non-volatile type or a mixture of volatile and non-volatile type memories.

[0011] Another embodiment of the present invention provides a service providing method for connecting a plurality of service providing systems to a service access apparatus via a communication network so as to enable each of those service providing systems to provide various services to the service access apparatus. The method has the steps of: sending display data including an indication for a service request to a second service providing system when a first service providing system provides a service to the service access apparatus; specifying a service request to be issued to the second service providing system in the display data displayed on the screen of the service access apparatus so as to enable the service access apparatus to request a service of the second service providing system, for example, the indication is selected by the user; starting communication between the service access apparatus and the second service

providing system via the first service providing system so as to provide the service of the second service providing system to the service access apparatus.

[0012] One embodiment of the present invention enables the first service providing system to hold a preset anonymous user ID used to log in the second service providing system, so that the first service providing system, when the service access apparatus issues a service request to the second service providing system, establishes communication with the second service providing system with use of the anonymous user ID.

[0013] Another embodiment of the present invention provides a service providing system for providing various services to a service access apparatus via a communication network. The service providing system has: a module for sending display data including a user selection for a service request from a second service providing system when it provides a service to the service access apparatus; a module for establishing communication with the second service providing system when the display data displayed on the screen of the service access apparatus is used to specify a service request to be issued to the second service providing system; and a module for relaying the communication between the second service providing system and the service access apparatus so as to enable the second service providing system to provide the service to the service access apparatus.

[0014] The above embodiments of the service providing system may further include: a module for holding a preset anonymous user ID used to log in the second service providing system; and a module for establishing the communication with the second service providing system with use of the anonymous user ID when the service access apparatus issues a service request to the second service providing system.

[0015] In one embodiment of the present invention a distributed system provides a plurality of services to a plurality of users. The system includes: a central server for providing a common access point for the plurality of users to the plurality of services; a plurality of area servers coupled to said central server for providing information on local services of said plurality of services to local users of said plurality of users; and a plurality of local cooperated company servers having said local services and coupled to said plurality of area servers. The process includes a local user accessing the central server with a user request for a local service. The central server sends the request to a area server within the users area, for example, telephone area code, and the area server obtains the local service for said local user. The area server has a mapping of local anonymous users to local users for access to local cooperated company servers having local services. The area servers are franchise servers of the central server. Thus, users have a common access point, hence one user ID and password to remember, but have access to common and local services. In addition the local service provider, for example, a small local business, may easily reach his/her local customer base.

[0016] These and other embodiments of the present invention are described in more detail in conjunction with the text below and attached figures.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] **FIG. 1** is a block diagram of a service providing system in an embodiment of the present invention.

[0018] **FIG. 2** is a block diagram of a service providing system of a cooperated company in an embodiment of the present invention.

[0019] **FIG. 3** is block diagram of a service access apparatus in an embodiment of the present invention.

[0020] **FIG. 4** is a block diagram of a network in one embodiment of the present invention.

[0021] **FIG. 4-1** is a block diagram of a network in an another embodiment of the present invention.

[0022] **FIG. 5** is a configuration of user information stored in a data memory of a service providing system of an embodiment of the present invention.

[0023] **FIG. 6** is a configuration of service information classified by user, stored in the data memory of the service providing system of an embodiment of the present invention.

[0024] **FIG. 7** is a configuration of substitute user information stored in the data memory of the service providing system of an embodiment of the present invention.

[0025] **FIG. 8** is a configuration of common information of state, stored in the data memory of the service providing system of an embodiment of the present invention.

[0026] **FIG. 9** is a configuration of information of state classified by area, stored in the data memory of the service providing system of an embodiment of the present invention.

[0027] **FIG. 10** is user information of a cooperated company stored in the data memory of the service providing system of a cooperated company of an embodiment of the present invention.

[0028] **FIG. 11** is a display example (1) of a service access screen.

[0029] **FIG. 12** is a display example (2) of the service access screen.

[0030] **FIG. 13** is a display example (3) of the service access screen.

[0031] **FIG. 14** is a display example (4) of the service access screen.

[0032] **FIG. 15** is a display example (5) of the service access screen.

[0033] **FIG. 16** is a display example (6) of the service access screen.

[0034] **FIG. 17** is a display example (7) of the service access screen.

[0035] **FIG. 18** is an example of a processing flow of an embodiment of the present invention.

[0036] **FIG. 19** is a flowchart of the management of common service.

[0037] **FIG. 20** is a flowchart of the management of service classified by area of an embodiment of the present invention.

[0038] FIG. 21 is a flowchart of the substitute replay processing of an embodiment of the present invention.

[0039] FIG. 22 is a flowchart of a synchronization processing of an embodiment of the present invention.

[0040] FIG. 23 is a flowchart of the management of common service announcement of an embodiment of the present invention.

[0041] FIG. 24 is a flowchart of the management of service announcement classified by area of an embodiment of the present invention.

[0042] FIG. 25 is a flowchart of the management of service classified by area, executed by the service providing system of cooperated company 201 of an embodiment of the present invention.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

[0043] Hereunder, the preferred embodiments of the present invention will be described with reference to the accompanying drawings.

[0044] At first, a description will be made for a configuration of a network as shown in FIG. 4 (to be described later) in which: a service providing system 101; a service providing system 201 of a cooperated company; and a service access apparatus 301 are connected to a communication network 400.

[0045] FIG. 1 is a block diagram of the service providing system 101. FIG. 2 is a block diagram of the service providing system 201 of a cooperated company. FIG. 3 is a block diagram of the service access apparatus 301, for example, a user's personal computer (PC), or a cell phone, or a Personal Digital Assistant (PDA).

[0046] Hereinafter, the configuration of the service providing system 101 will be described with reference to FIG. 1. The service providing system 101 provides various services to the users via a communication network.

[0047] The service providing system 101 includes a CPU 111; a memory 112; a communication block 113; a data transfer path 115; a program memory 116; and a data memory 117. The CPU 111 is a central processing unit for controlling the whole service providing system 101. The memory 112 is a storage device for storing various processing programs loaded from a non-volatile program memory 116 so as to execute them and storing other various data. The communication block 113 is an interface connected to a communication line 114. The communication block 113 enables the service providing system 101 to communicate with other systems and apparatuses via a communication network. The data transfer path 115 is a bus line connected to each of the components.

[0048] The program memory 116 stores various processing programs to be executed by the CPU 111. In particular, the program memory 116 stores the programs for: management of common service 121; management of service classified by area 122; substitute replay processing 123; synchronization processing 124; management of common service announcement 125; and management of service announcement classified by area 126. Each of those programs is loaded into the memory 112 when it is executed. In the following description, however, a program loaded into the memory 112 and being executed is also referred to as a number within 121 to 126. The data memory 117 is a secondary storage device for storing various data to be used by the service providing system 101. In particular, the data memory 117 stores items of: user information 131; service information classified by user 132; substitute user information 133; common information of state 134; and information of state classified by area 135. Each processing program stored in the program memory 116 and each information stored in the data memory 117 will be detailed later.

[0049] The configuration of the service providing system of a cooperated company 201 will be described with reference to FIG. 2. The service providing system of cooperated company 201, just like the service providing system 101, provides various services to the users, especially via the service providing system 101.

[0050] The service providing system of cooperated company 201 includes: a CPU 211; a memory 212; a communication block 213; a data transfer path 215; a program memory 216; and a data memory 217. The CPU 211 is a central processing unit for controlling the whole service providing system 201. The memory 212 is a storage device for storing various processing programs loaded from a program memory 216, which is a secondary storage device, so as to execute them and storing various other data. The communication block 213 is an interface connected to a communication line 214. The communication block 213 enables the service providing system of cooperated company 201 to communicate with other systems and apparatuses via a communication network. The data transfer path 215 is a bus line connected to each of the components.

[0051] The program memory 216 is a secondary storage device for storing various processing programs to be executed by the CPU 211, and particularly stores a program for management of service of cooperated company 221. The program is loaded into the memory 212 when it is executed. In the following description, however, the program loaded into the memory 212 and being executed is also referred to as a number of 221. The data memory 217 is a secondary storage device for storing various data to be used by the service providing system of cooperated company 201, and particularly stores user information of cooperated company 231. Each processing program stored in the program memory 216 and each information stored in the data memory 217 will be detailed later.

[0052] The configuration of the service access apparatus 301 will be described with reference to FIG. 3. The service access apparatus 301 is used by each user so as to access his/her desired services.

[0053] The service access apparatus 301 includes: a CPU 311; a memory 312; a communication block 313; a data transfer path 315; a program memory 316; a display block 317; and an input block 318. The CPU 311 is a central processing unit for controlling the whole service providing system 301. The memory 312 is a storage device for storing various processing programs loaded from a program memory 316, which is a secondary storage device, so as to execute them and storing various other data. The communication block 313 is an interface connected to a communication line 314. The communication block 313 enables the service access apparatus 301 to communicate with other

systems and apparatuses via a communication network. The data transfer path **315** is a bus line connected to each of the components.

[0054] The program memory **316** is a secondary storage device for storing various processing programs to be executed by the CPU **311**, and particularly stores a service access program **321**, for example, a Web browser, such as Netscape® or Microsoft® Internet Explorer. In the following description, the service access program loaded into the memory **312** and being executed is also referred to as a number of **321**. The display block **317** displays various information according to each instruction received from the CPU **311**. The input block **318** is such an input device as a keyboard and a mouse used by the user to enter various data items. The program stored in the program memory **316** will be detailed later.

[0055] **FIG. 4** is a block diagram of a network used by the systems in this embodiment. The service providing system **101**, the service providing system of cooperated company **201**, and the service access apparatus **301** are connected to the communication network **400**. The communication network **400** is, for example, the Internet.

[0056] Each user, who is a customer of services provided by the systems in this embodiment can access the service providing system **101** by executing the service access program **311**, e.g. browser, in his/her service access apparatus so as to receive a desired service therefrom. In this case, it is premised that the user has already obtained a user code (user ID) and a password in response to his/her personal information sent to and registered in the service providing system **101** before he/she accesses the service therefrom **101**.

[0057] The service providing system **101** executes the management of common service **121**, thereby providing a requested common service to the user. A common service is, for example, a service provided to all the users commonly; it is one of the services provided to the users from the service providing system **101**. For example, the common service is a service for providing travel information to a user and accepting an application for the travel from the user. The service providing system **101** also executes the management of service classified by area **122**, thereby providing a service classified by area to the user. The service classified by area means a service provided only to the users living in a specified area (ex., a residential area). For example, the service classified by area includes taxi reservation in a subject area, product buying in a shop opened for business in the subject area.

[0058] The management of common service **121** and the management of service classified by area **122** are synchronized with each other by a synchronization processing **124**. The synchronization processing means matching between information of state when a service is provided to a user by the management of common service **121** and information of state when a service is provided to the user by the management of service classified by area **122**. This processing will be described more in detail later.

[0059] The system in this embodiment further enables the user of a service access apparatus **301**, which is a customer of the system, to use the services of a company cooperated with the service providing company that operates this service providing system **101** in addition to the services provided by the service providing system **101**. The service providing system of cooperated company **201** is an apparatus that provides the services of the cooperated company. The service providing company that runs the service providing system of cooperated company **201** (hereinafter to be referred to as a cooperated service providing company) and the service providing company that runs the service providing system **101** are associated with each other under a contract. The users of the service providing system **101** can thus access the service providing system of cooperated company **201** via the service providing system **101**. Specifically, the service access program **311** of the service access apparatus **301** can access the service providing system of cooperated company **201** via the substitute reply processing **123** of the service providing system **101** so as to use various services of the cooperated company provided via the management of service of cooperated company **221**. At this time, the user is not required to register himself/herself in the service providing system of cooperated company **201** so as to obtain an ID; the user can receive a service of the cooperated company just like a service provided from the service providing system **101** without knowing the difference.

[0060] **FIG. 4-1** shows a network block diagram of an alternate embodiment of the present invention. The central server **101-1** has many of the functions of the service providing system **101** of **FIG. 4**, including the management of common service **121**, the substitute reply processing **123**, and synchronization processing **124**. However, the management of service classified by area **122** has been decentralized into local servers by area, for example, Area1 Server **122-1** and Area2 Server **122-2**. Central server **101-1** is connected to Area1 Server **122-1** and Area2 Server **122-2** via main network **400-1**. The area servers are franchise servers of the central server and provide local area Web page content and services. For example, Area1 Server **122-1** is connected to User1**301-1**, user2**301-2** and Coop Co. **2** Server **201-2** via area network **400-2**. Area2 Server **122-2** receives services from Coop co. **3** server **201-3** and provides those services to user3**301-3** via area2 network **400-3**. The users, e.g., users **301-1**, **301-2**, and **301-3**, still log on to the central server **101-1** with their user ID and password for access to all services they are entitled to, but in this embodiment the area server is the Web server for the user, i.e., both local content and content from the central server **101-1** is sent, for example, from area**1** server **122-1** to user1's **301-1** Web browser via area**1** network **400-2**. The users (**301-1**, **301-2**, and **301-3**) are similar to the service access apparatus **301**, and the Coop Co. Servers **201-1**, **201-2**, and **201-3** are similar to the service providing system of cooperated company **201** of **FIG. 4**. Coop Co. **1** Server **201-1** gives an example of a cooperated (Coop) company (Co.) service that is provided via the main network **400-1**, for example a travel service, while Coop Co. **2** Server **201-2** gives an example of a local service, such as, from a local taxi company.

[0061] **FIG. 5** is a configuration of user information **131** stored in the data memory **117** of the service providing system **101**. The user information **131** is a table for storing various information items of a user who can receive services provided by the service providing system **101**. The user information table **131** includes fields of user code **501**, password **502**, area code **503**, account holder code **504**, and account holder name **505**.

[0062] The user code **501** is an ID used to identify a user. The password **502** is needed by the user to access the service providing system **101**. The area code **503** is used to identify an area to which the user belongs. The account holder code **504** identifies an account holder who pays for a service provided from the service providing system **101** as needed. The account holder name **505** is the name of the account holder, for example. A user who wants to receive a service from the service providing system **101**, before using the service actually, must make an application to the subject service providing company that runs the service providing system **101** and his/her personal information is registered in the user information table **131**.

[0063] **FIG. 6** is a configuration of the service information classified by user **132** stored in the data memory **117** of the service providing system **101**. The service information classified by user **132** is a table for storing information denoting services accessible, i.e., allowed for use by the user. The service information classified by user **132** includes fields of user code **601**, service code **602**, service name **603**, service user code **604**, and service user name **605**.

[0064] The user code **601** is used to identify a user. The service code **602** and the service name **603** are used to identify a service usable by the user. The service user code **604** and the service user name **605** are used to denote information on the service provided. For example, because SA**00** denotes "Common"*6a* for the service providing company name **605**, the service is a common one provided to all the users. SA**01** denotes that the subject service is a service classified by area, which is provided only to the users in the Tokyo area *6b*. SX**00604***a* denotes that the subject service is provided by the cooperated company X**006***c*. Those information items are set for a user when the user is registered in the user information **131** (or later).

[0065] **FIG. 7** is a configuration of the substitute user information **133** stored in the data memory **117** of the service providing system **101**. The substitute user information **133** is a table for storing items of ID and password allocated to a user and used to access the service providing system of cooperated company **201** associated with the service providing system **101** via the service providing system **101**. The substitute user information **133** includes fields of substitute user code **701**, substitute password **702**, cooperated company code **703**, and user code **704**.

[0066] The substitute user code **701** is a user code (ID) allocated automatically to a user when the user accesses the service providing system of cooperated company **201** through a substitute replay processing **123** executed by the service providing system **101**. The substitute password **702** is a password corresponding to the substitute user code **701**. The cooperated company code **703** is used to identify the service providing system of cooperated company **201** to be accessed with use of the substitute user code **701** and the substitute password **702**. The user code **704** identifies a user who is accessing the service providing system of cooperated company **201**. For example, user**01704***a* may want to access Cooperated Company X**00703***a* (*6c* in **FIG. 6**), i.e., Air Ticket Reservation **603***a* service with Service Code SX**00604***a*. Service providing System **101** logs on to the service providing system of cooperated company X**00703***a* using substitute user code Xuser**01701***a* and the substitute password Xpws**01702***a*.

[0067] In this embodiment, the number of users who can access the service providing system of cooperated company **201** concurrently via the service providing system **101** is decided beforehand under the contract made between the service providing company and the cooperated company **201**. The user codes (ids) and passwords for those decided number of users used to access the service providing system of cooperated company **201** are given to the service providing company **101** by the cooperated company **201**. The user ids are essentially a set of unassigned or anonymous user ids which may be used to login to the service providing system of cooperated company **201** by the service providing system **101**. The service providing system **101** will do the mapping of these unassigned or anonymous user ids (codes), and also the unassigned user associated passwords, in the table **133** in **FIG. 7**. The service providing company **101** thus sets received information in the substitute user code **701** and the substitute password **702** of the substitute user information table **133**. The user code **704** is kept emptied until the user requests an access to the service providing system of cooperated company **201** so that the user can access the cooperated company with use of the corresponding substitute user code **701** and password **702** so as to use a service thereof.

[0068] **FIG. 8** is a configuration of the common information of state **134** stored in the data memory **117** of the service providing system **101**. The common information of state **134** is a table for storing information related to the services (including not only common services, but also services classified by area and by cooperated company) used by all the users. The common information of state **134** includes fields of user code **801**, service code **802**, cooperated company user code **803**, state code **804**, and service providing time **805**.

[0069] The user code **801** is used to identify a user who uses a service. The service code **802** is used to identify a service used by the user. The user code of cooperated company **803** is used to store a substitute user code (**701** in **FIG. 7**) used to access the service providing system of cooperated company **201** when the user uses a service of a cooperated company provided via the service providing system of cooperated company **201**. For example user**01801***a* (user**017***a* in **FIG. 6**) has service code svc**03802***a* (svc**03602***a* in **FIG. 6**) and cooperated company user code Xuser**01803***a* (substitute user code Xuser**01701***a* of **FIG. 7**). The state code **804** stores the various states that occur when the user uses a service. For example, the state code **804** stores such states as login and logout, or start and end. The service providing time **805** stores a time at which the subject state occurs.

[0070] **FIG. 9** is a configuration of the information of the state classified by area **135** stored in the data memory **117** of the service providing system **101** of an embodiment of the present invention. The information of the state classified by **135** is set for each area. In the information of state classified by area **135** is stored with respect to services (including not only services classified by area, but also common services and services of cooperated companies) used by the users who live in the subject area. The information of state classified by area **135** includes fields of area code **901**, user code **902**, service code **903**, state code **904**, and service providing time **905**.

[0071] The user code **902** identifies a user. The area code **901** identifies an area to which the user belongs. The service code **903** stores a code for identifying a service used by the user. The state code **904** stores various states to occur when a service is used. The service providing time **905** stores a time at which the state occurs.

[0072] **FIG. 10** is a configuration of the user information of cooperated company **231** stored in the data memory **217** of the service providing system **101**. The user information of cooperated company **231** is a table for storing preset items of user code and password with which the service providing system of cooperated company **201** accepts an access. The user information of cooperated company **231** includes fields of user code of cooperated company **1001**, password of cooperated company **1002**, account holder code of cooperated company **1003**, and account holder name of cooperated company **1004**.

[0073] The user code of cooperated company **1001** (corresponding to **701** in **FIG. 7**) that was sent by cooperated company **201** in order to access the cooperated company **201** system is stored by service providing system **101**. The password of cooperated company **1002** corresponding to the user code is also stored in memory **217**. The account holder code of cooperated company **1003** and the account holder name of cooperated company **1004** store a code and a name used to identify the account holder corresponding to the user code of cooperated company **1001** and the password of cooperated company **1002**. The account holder is, for example, a person who pays for a service of the service providing system of cooperated company **201**. However, because the accounting generated when the user uses a service of the service providing system of cooperated company **201** via the service providing system **101** is charged to the original requester of the user IDs and passwords, the account holder is not an individual user **301**, but the service providing system **101**. The service providing system **101** then passes on the service fee from the cooperated company **201** to the individual user **301**. The service providing system **101** may optionally charge the user **301** a small fee for this pass-through service.

[0074] **FIGS. 11 through 17** show examples of the service access screen displayed at the system in this embodiment. **FIG. 18** shows an example of the processing order of management of service providing in the system in this embodiment. Hereinafter, a description will be made for each screen displayed on the display block **317** of the service access apparatus **301** when processings are executed sequentially in the order shown in **FIG. 18**.

[0075] In **FIG. 18**, the service access program **321**, for example an Internet browser, that runs in the service access apparatus **301**, for example a personal computer or cell phone or Personal Digital Assistant (PDA), operated by the user, when the user requests a common service, issues the service request to the management of common service **121**(Sa**10**). In response to this request, the management of common service **121** displays the screen **1110** shown in **FIG. 11** on the service access screen **1101** of the service access apparatus **301** (Sa**20** in **FIG. 18**). This screen **1110** is a travelers' site **A00** login screen. The screen **1110** has a user code input field **1111**, a password input field **1112**, and a button **1113** for logging in the travel plan screen (service code: svc**01**). On this screen **1110**, the user is requested to enter his/her user code and password (those registered in the user information **131** in **FIG. 5**) in the user code input field **1111** and the password input field **1112**, then press the log-in button **1113**.

[0076] The service access program **321** then sends the user code and the password entered by the user to the management of common service **121**(Sa**30** in **FIG. 18**). Checking the user code and the password, the management of common service **121** displays the screen svc**01** for starting the common service (Sa**40** in **FIG. 18**).

[0077] **FIG. 12** shows an example of the screen svc**01** for starting the common service to be displayed on the display block **317** of the service access apparatus **301**. On the service access screen **1101** is displayed a field **1210** received from the management of common service **121**; field **1210** is used for logging in travel plan. In addition to an advertisement message of travel plan recommended to the user, the field **1210** also displays a RESERVE button **1211** used to reserve a travel plan and a CANCEL button **1212**.

[0078] The service access screen **1101** always displays the original service access field, as well as a common service announcement field and a service announcement field classified by area. In one embodiment the screen **110** (**FIG. 11**) and screen **1101** (**FIG. 12**) is has information sent directly from the central server **101-1** to the user, for example, user **301-1**. In an alternate embodiment the screen information from the central server **101-1** is sent indirectly through, for example, the area1 server **122-1** to the user**1301-1**.

[0079] In **FIG. 12**, numeral **1220** denotes the common service announcement field and **1230** denotes the service announcement field classified by area. The common service announcement field **1220** displays various announcement messages on common services including the information about a common service expected as the next service to be accessed by the user, according to the state of the service execution up to that time. The service announcement field classified by area **1230** displays various announcement messages on services classified by area, including the information about a service classified by area expected as the next service to be accessed by the user. The common service announcement field **1220** and the service announcement classified by area **1230** may display, for example, a newly set frame and a newly opened window with use of the service access program (browser) that runs in the service access apparatus **301**. In an alternative embodiment the information for the service announcement classified by area **1230** comes directly from the area server, for example, area1 server **122-1**.

[0080] Assume that after that, the user has pressed the RESERVE button **1211** on the screen shown in **FIG. 12** so as to reserve the recommended travel plan. The system thus accepts the input done on the screen svc**01** (the reservation for travel to Hawaii in this case)(Sa**50** in **FIG. 18**). Ending this reservation processing, the management of common service **121** sends an end notice of the service svc**01** (this travel plan) to the service access program **321** (Sa**60**).

[0081] **FIG. 13** shows an example of a screen for ending a common service. The screen is sent in Sa**60**. On the service access screen **1101** is displayed the ending service access field **1310** for the travel plan. This field **1310** displays a message "Thank you for accessing" when the service is

ended, as well as a LOGOUT button **1311** and a RESERVE button **1312** (service code: svc**02**) used for taxi reservation, which is considered to be necessary for the travel plan accepted at that time. The common service announcement field **1320** displays a common service advertisement message for the service recommended to the user and the service announcement field **1330** displays an advertisement message of a service classified by area, recommended to the user. In particular, the service classified by area **1330** displays a button (service code: svc**02**) **1331** for calling the reservation screen for taxi reservation expected to be needed in the travel just after a travel plan reservation service is executed.

[**0082**] Assume that the user has pressed this RESERVE button **1312** (or **1331**) for taxi reservation, which is one of the services classified by area recommended on the screen shown in **FIG. 13**. Consequently, the service access program **321** sends a service request for service code svc**02** (a service available only in that area) to the management of service classified by area **122**(Sb**10** in **FIG. 18**), for example Coop Co.**2** Server **201-2** in **FIG. 4-1**. In response to the request, the management of service classified by area **122** displays the screen svc**02** for taxi reservation (Sb**20**).

[**0083**] **FIG. 14** shows an example of a field **1410** displayed on the display block **317** of the service access apparatus **301**. The service access program **321** receives the field **1410** from the management of service classified by area **122** in Sb**20**. The field **1410** for taxi reservation (service code: svc**02**) is displayed on the service access screen **1101**. This field **1410** displays the date and time on which a taxi reservation is made, as well as places to get on and off according to the travel plan reservation made by the user. The date and time, as well as the places to get on and off are set automatically on the field **1410**. The field **1410** also displays a RESERVE button **1411** and a CANCEL button **1412**. The common service announcement field **1420** displays an advertisement message about a common service expected as the next service to be accessed by the user. In the same way, the service announcement field **1430** classified by area displays an advertisement message about a service specific to the area and expected as the next service to be accessed by the user.

[**0084**] To reserve a taxi on the screen **1410** shown in **FIG. 14**, the user presses the RESERVE button **1411**. Assume that the user has pressed the RESERVE button **1411**. Then, the service access program **321** sends the command that the RESERVE button **1411** has been pressed, to the management of service classified by area **122**(Sb**20** in **FIG. 18**). In response to the command, the management of service classified by area **122** then processes the taxi reservation. When the processing is ended, the management of service classified by area **122** sends the service ending screen to the service access program **321**(Sb**40**).

[**0085**] **FIG. 15** shows an example of a field **1510** displayed on the display block **317** of the service access screen **301**. The service access program **321** receives the screen **1510** from the management of service classified by area **122** in Sb**40**. The field **1510** appears on the service access screen **1101** when the taxi reservation is ended. The field **1510** displays a LOGOUT button **1511**, and a svc**03** button **1512** for calling an air ticket reservation service (service code: svc**03**), which is expected as the next service to be accessed by the user. The common service announcement field **1520** and the service announcement field classified by area **1530** display advertisement messages about services expected as the next items to be accessed by the user respectively. The air ticket reservation service svc**03** recommended to the user is a service of a cooperated company. The service is displayed in the common service announcement field **1520** at this time, since there is no need to let the user know how the service is provided to the user.

[**0086**] Assume that the user has pressed the reserve button **1512** so as to reserve an air ticket on the field **1510** shown in **FIG. 15**. Then, the service access program **321** sends the command that the button **1512** has been pressed so as to access the air ticket reservation, to the substitute reply processing **123**(Sc**10** in **FIG. 18**). In response to the command, the substitute reply processing **123** relays the access to the management of service of cooperated company **221**(Sc**20**). In response to the request, the management of service of cooperated company **221** sends the air ticket reservation screen (service code: svc**03**) to the substitute reply processing **123**(Sc**30**). The substitute reply processing **123** relays the screen and sends it to the service access program **321**(Sc**40**), so that the air ticket reservation screen is displayed on the display block **317** of the service access apparatus **301**.

[**0087**] **FIG. 16** shows an example of a service access screen **1101** for the air ticket reservation service, which is one of the services of a cooperated company received such way. The service access screen **1101** displays an air ticket reservation access filed **1610** in which a reserve button **1611** and a cancel button **1612** are displayed. The destination, as well as the date and time for the departure of the flight to be reserved are selected optimally according to the travel plan reserved previously by the user and displayed there. The user is then requested to press the reserve button **1611** to reserve the flight and press the cancel button **1612** to stop the flight reservation service. The common service announcement field **1620** and the service announcement field classified by area **1630** display advertisement messages about the services expected as the next services to be accessed by the user respectively.

[**0088**] Assume that the user has pressed the reserve button **1611** to reserve the recommended flight. Then, the service access program **321** sends the command that the button **1611** has been pressed, to the management of service of cooperated company **221**(Sc**50** and Sc**60** in **FIG. 18**). In response to the command, the management of service of cooperated company **221** processes the flight reservation. When the reservation is ended, the management of service of cooperated company **221** sends the ending screen for the service of a cooperated company to the service access program **321** via the substitute reply processing **123** (Sc**70** and Sc**80**).

[**0089**] **FIG. 17** shows an example of the service access screen **1101** to appear when this service of the cooperated company is ended. On the service access screen **1101** are displayed the flight reservation ending field **1710** in which a logout button **1711** is displayed. The common service announcement field **1720** and the service announcement field classified by area **1730** display information related to the reservation made by the user, as well as advertisement messages about services expected as the next services to be accessed by the user respectively.

[0090] A description will be made for the processing procedures executed in the system in this embodiment (in particular, those described with reference to FIGS. **11** through **18**) with reference to the flowcharts shown in **FIGS. 19 through 25**. Assume that the same symbols (Sa**10**, Sa**20**, etc.) are given to the same steps in **FIGS. 19 through 25** as those shown in **FIG. 18**.

[0091] **FIG. 19** is a flowchart of the management of common service **121** executed by the service providing system **101**. At first, the management of common service **121** executes a processing for accepting a service request issued by the user with use of the service access program **321** of the service access apparatus **301** in step Sa**10**. In response to the request, the system displays the login screen shown in **FIG. 11** in step Sa**20**. The system then accepts such information as the user code and the password entered by the user on the login screen in step Sa**30**. Then, the system reads the user information **131** described with reference to **FIG. 5** in step Sa**31** so as to check whether or not the user code and the password entered by the user in step Sa**32-1** are identical to those registered in the user information **131**. Where they are illegal, the system stops the processing and control returns to the previous step.

[0092] When the user code and the password are correct, the system reads the service information classified by user **132** described in reference to **FIG. 6** in step Sa**33**. After that, the system checks whether or not the service code specified by the user at the login time is registered in the service information classified by user **132** in step Sa**32-2** (that is, whether or not the user is enabled to use the service). When the service code is not registered, the user is not enabled to use the service. The system thus stops the processing and returns to the previous step. Where the service code is registered in the information **132**, control goes to step Sa**34**.

[0093] In step Sa**34**, the system writes the effect that the user has logged in the processing, in the common information of state **134**. In the example, the effect is written just like the row data **8***a* shown in **FIG. 8**. After that, the system displays the service code svc**01** screen shown in **FIG. 12** in step Sa**40**. In step Sa**45**, the system writes the effect that the svc**01** service has been started in the common information of state **134** just like the row data **8***b* in **FIG. 8**. Then, the system accepts the information entered by the user on the screen shown in **FIG. 12** in step Sa**50**. It is assumed here that the user has pressed the reserve button **1211** so as to process the reservation. When the reservation is ended, the system displays the svc**01** service ending screen in step Sa**60**. In this case, the service ending screen shown in **FIG. 13** is displayed. After that, the system writes the svc**01** service ending information in the common information of state **134** just like the row data **8***c* in **FIG. 8**.

[0094] **FIG. 20** is a flowchart of the management of service classified by area **122** executed by the service providing system **101**. At first, the system accepts a request for a service classified by area from a user in step Sb**10**. The system then reads the information of state classified by area **135** shown in **FIG. 9** in step Sb**11** and takes over the user information in step Sb**12**. Taking over user information means a processing for obtaining user information read in the previous step Sa**31**. Then, the system reads the service information classified by user **132** shown in **FIG. 6** in step Sb**13** and checks whether or not the service code specified by the user is registered in the service information classified by user **132** in step Sb**14**. Where the service code is not registered, the user is not enabled to use the service. The system thus stops the processing and returns to the previous step.

[0095] Where it is decided in step Sb**14** that the service code is registered, the system displays the screen for the specified service classified by area in step Sb**20**. In this example, the taxi reservation screen svc**02** shown in **FIG. 14** is displayed. After that, the system writes the result that the service has been started in the information of state classified by area **135** shown in **FIG. 9** in step Sb**21**. In this example, the row data **9***d* in **FIG. 9** is written. The information written in the common information of state **134** shown in **FIG. 8** is reflected in the information of state classified by area **135** shown in **FIG. 9** by a synchronization processing **124** to be described later with reference to **FIG. 22**. The information written in the information of state classified by area **135** shown in **FIG. 9** is reflected in the common information of state **134** shown in **FIG. 8**. This means that the information items **9***a* to **9***c* related to the previously executed common service are already written there when row data **9***d* is written.

[0096] The system then accepts an input on the screen for the service classified by area in step Sb**30**. It is assumed here that the user has pressed the reserve button **1411** shown in **FIG. 14** so as to reserve a taxi. When the taxi reservation is ended, the system displays the ending screen of the management of service classified by area **122** in step Sb**40**. In this example, the ending screen shown in **FIG. 15** is displayed. Then, the system writes the row data **9***e* denoting the effect that the svc**02** service has been ended, in the information of state classified by area **135** shown in **FIG. 9** in step Sb**41** and returns to the previous step.

[0097] **FIG. 21** is a flowchart of the substitute reply processing **123** executed by the service providing system **101**. Accepting the service request (for a service of a cooperated company) from the user in step Sc**10**, the system reads the common information of state **134** shown in **FIG. 8** in step Sc**11**, then takes over the user information in step Sc**12**. After that, the system reads the user's service information classified by user in step Sc**13** and checks whether or not the service of the cooperated company specified by the user is registered in the service information classified by user **132** in step Sc**14**. When the service code is not registered, the user is not enabled to use the service. The system thus stops the processing and returns to the previous step.

[0098] Where it is decided in step Sc**14** that the service code is registered, the system reads the substitute user information **133** shown in **FIG. 7** in step Sc**15**. The system then searches an empty user code field **704** in the substitute user information **133** in step Sc**16**. Where no such an empty substitute user code is found, the system stops the processing and returns to the previous step. Where an empty substitute user code is found, the system allocates the substitute user code in step Sc**17**. Specifically, the system writes the subject user code in the user code field **704** in the substitute user code row of the substitute user information **133** just like the row data **704***a* in **FIG. 7**.

[0099] Then, the system takes over the service access in step Sc**20**. This is a processing for obtaining the login screen from the service providing system of cooperated company

201. In step Sc21, the system inputs data on the login screen for the service providing system of cooperated company 201. This is a processing for logging in the service providing system of cooperated company 201 automatically with use of the substitute user code and the substitute password corresponding to the service code allocated respectively in step Sc17. In response to this login, the specified service providing screen is sent to the system from the service providing system of cooperated company 201. The system thus relays the screen data (1610 in FIG. 16) to the service access apparatus 301 in step Sc40. After that, the system writes the effect that the service providing system of cooperated company 201 has started the specified service, in the common information of state 134 shown in FIG. 8 in step Sc41.

[0100] The system then accepts an input from the service providing system of cooperated company 201 on the service screen in step Sc50. This is a processing for accepting an input on the screen for services of a cooperated company shown in FIG. 16. Accepted data is relayed in step Sc60 to the service providing system of cooperated company 201. When the cooperated company's service is ended, the system relays the end notice to the service providing system of cooperated company 201 in step Sc70 and displays the ending screen in step Sc80. In this case, the screen 1710 for ending the cooperated company's service shown in FIG. 17 is relayed to and displayed on the screen of the service access apparatus 301. After that, the system writes the effect that the service has been ended, in the common information of state 134 shown in FIG. 8 just like the row data 8*g* and 8*h*, then returns to the previous step.

[0101] FIG. 22 is a flowchart of the synchronization processing 124 executed by the service providing system 101. This processing 124 is always repeated at predetermined intervals. At first, the system reads the common information of state 134 shown in FIG. 8 in step Sd10, then checks whether or not the state code 804 is changed in step Sd20. Where the state code is not changed, control returns to the previous step. Where the state code is changed (that is, new row data is added to the common information of state 134), the system reads the user information 131 in step Sd30, then obtains an area code 503 corresponding to the user code 801 of the row data added newly to the common information of state 134 in step Sd40. Then, the system adds the row data added newly to the common information of state 134 as described above to the information of state classified by area 135 together with the area code obtained in the previous step in the next step Sd50. Consequently, the data added to the common information of state 134 is affected in the information of state classified by area 135.

[0102] The processing's in steps Sd60 to Sd80 are executed so as to reflect the data added to the information of state classified by area 135 in the common information of state 134. Specifically, the system reads the information of state classified by area 135 in step Sd60, then checks whether or not the state code is changed (whether or not any new row data is added to the information of state classified by area 135) in step Sd70. Where the state code 904 is not changed, the system returns to the previous step with no operation. Where the state code 904 is changed, the system makes the row data added to the information of state classified by area 135 reflect in the common information of state 134 in step Sd80, then returns to the previous step.

[0103] Because both of the common information of state 134 and the information of state classified by area 135 are stored in the service providing system 101 in this embodiment, the processings in steps Sd10 to Sd50 for reflecting the common information of state 134 in the information of state classified by area 135 are combined with the processings in steps Sd60 to Sd80 for reflecting the information of state classified by area 135 in the common information of state 134 into a series of processing's as shown in FIG. 22. Where the common information of state 134 and the information of state classified by area 135 are stored in different apparatuses, however, those processings must be executed separately in those different apparatuses.

[0104] FIG. 23 is a flowchart of the management of common service announcement 125 executed by the service providing system 101. The common service announcement processing 125 is always repeated at predetermined intervals. At first, the system reads the common information of state 134 in step Se10, then checks whether or not the state code 804 is changed in step Se20. Where the state code 804 is not changed, the system returns to the previous step with no operation. Where the state code 804 is changed, the system generates screen data classified by state in step Se30, then displays the common service announcement screen in step Se40 and returns to the previous step. The screen data classified by state means screen data that should be announced to the user according to the state in the past or at that time. For example, because the common information of state 134 indicates that the user has used the reservation service for travel to Hawaii, the system displays a common service announcement screen for recommending an air ticket reservation service corresponding to the previous service.

[0105] FIG. 24 is a flowchart of the management of service announcement classified by area 126 executed by the service providing system 101. The management of service announcement classified by area 126 is always repeated at predetermined intervals. At first, the system reads the information of state classified by area 135 in step Sf10, then checks whether or not the state code is changed in step Sf20. When the state code is not changed, control returns to the previous step. When the state code is changed, the system generates state screen data in step Sf30 and displays the service announcement screen classified by area in step Sf40. The system then returns to the previous step. The screen data classified by state means screen data that should be announced to the user according to the state in the past or at that time. For example, on the service access field shown in FIG. 16 is displayed a service announcement field classified by area, which recommends the user to buy some beach goods, since the information of state classified by area 135 indicates that the user has reserved a travel to Hawaii.

[0106] FIG. 25 is a flowchart of the management of service of cooperated company 221 executed by the service providing system of cooperated company 201. At first, the system accepts a service request from a user in step Sg10. Then, the system displays the login screen in step Sg20 and accepts an input done on the login screen in step Sg30. After that, the system reads the user information of cooperated company 231 in step Sg40 and checks the user code and the password of the user in step Sg41. Where the user code and the password are illegal, the system stops the processing and returns to the previous step. Where the user code and the password are correct, the system displays a service screen

(ex., svc**03** in **FIG. 16**) in step Sg**50**. The system then accepts an input done on the screen in step Sg**60**. When the processing ends, the system displays the ending screen in step Sg**70** and returns to the previous step.

[**0107**] At this time, the service providing system of cooperated company **201** is not requested to distinguish the user's access between indirect one via the service providing system **101** and direct one from the user's own service access apparatus **301**. This is because the input on the login screen is done in the substitute replay processing on behalf of the user as described with reference to **FIG. 21** even when the user's access is done indirectly via the service providing system **101**.

[**0108**] In one embodiment, the description has been made for a system configured by a service providing system **101**, a service providing system of cooperated company **201**, and a service access apparatus **301** connected to the communication network **400** respectively. However, each of those systems and apparatuses may be plural in number. For example, each user may have a plurality of service access apparatuses **101** connected to the network **400** and each cooperated company may have a plurality of service providing system of cooperated company **201** connected to the network **400** in case where there are a plurality of cooperated companies. The service providing system **101** may be realized by distributed processing's executed by a plurality of systems connected to the communication network **400**. In this case, the processing's **121** to **126** shown in **FIG. 1** may be executed by different systems. The information **131** may also be stored in different systems. In particular, in case where a system for providing common services is separated from a system for providing services classified by area, the information of state classified by area **135** and the service information classified by user **132** should be stored in different systems classified by area.

[**0109**] Because the access state of every user is recorded in the common information of state **134**, the accounting of each user may be done according to the common information of state **134**. Because the access state of each user in each area can be known by the information of state classified by area, it is possible for those apparatuses to build a relationship between a headquarter and franchise shops. For example, it is possible to install a server used for services classified by area in each area so as to calculate the access state of each user in the area and return profits to the manager of the server accessed by many users from the headquarter.

[**0110**] As described above, according to an embodiment of the present invention, a user is requested to register his/her access to just one service access apparatus so as to receive a service at another service access apparatus via the service access apparatus. The user can thus be freed from troublesome procedures, since he/she is not requested to register his/her access to each service access apparatuses. The user can also omit management of his/her user ID and password for each service apparatus, thereby his/her labor is much reduced. Because the user's personal information is stored only in the service access apparatus he/she has registered his/her access, he/she is much freed from worry about the illegal use of his/her personal information by others. As a result, embodiments of the present invention can realize simple and high security services. Each company that runs a service providing system can also provide a variety of attractive services to get an advantage over other companies so as to obtain many more users in the market.

[**0111**] Another embodiment of the present invention provides a computer readable medium for storing code for providing use by a user of a plurality of services through a common access point. The code for providing said user access to a first computer system; code for said first computer system accessing a second computer system comprising said service in response to a user request for a service of said plurality of services; and code for providing said service to said user from said second computer system through said first computer system.

[**0112**] Although the above functionality has generally been described in terms of specific hardware and software, it would be recognized that the invention has a much broader range of applicability. For example, the software functionality can be further combined or even separated. Similarly, the hardware functionality can be further combined, or even separated. The software functionality can be implemented in terms of hardware or a combination of hardware and software. Similarly, the hardware functionality can be implemented in software or a combination of hardware and software. Any number of different combinations can occur depending upon the application.

[**0113**] Many modifications and variations of the present invention are possible in light of the above teachings. Therefore, it is to be understood that within the scope of the appended claims, the invention may be practiced otherwise than as specifically described.

What is claimed is:

1. A method for a user to use a plurality of services through a common access point, comprising:

providing said user access to a first computer system;

responsive to a user request for a service of said plurality of services, said first computer system accessing a second computer system, comprising said service; and

providing said service to said user from said second computer system through said first computer system.

2. The method of claim 1 wherein said second computer system is not provided with any information on said user, when said first computer system accesses said second computer system.

3. The method of claim 1 wherein said providing said user access to said first computer system, comprises giving said user a user code and a password.

4. The method of claim 1 farther comprising said second computer system giving said first computer system a group of user ID's and passwords for use by said first computer system in accessing said second computer system.

5. The method of claim 4 wherein said user ID's are anonymous user IDs.

6. The method of claim 1 wherein when said user request for said service is for a local service of said plurality of services located in a same area as said user, said first computer system accesses a local computer system having said local service.

7. A service providing system for providing a service located on a service computer system to a user computer system, wherein said user computer system need only access said service providing system to obtain said service and

wherein said service computer system is coupled to said service providing system via a communications network, said service providing system comprising:

a storage system;

an access table stored in a first part of said storage system, comprising a user identification for access to said service providing system by said user computer system;

a mapping table stored in a second part of said storage system, comprising said user identification and associated login information; and

responsive to a request by said user for said service, software stored in stored in a third part of said storage system, for accessing said service computer system using said associated login information and for obtaining said service for said user computer system.

8. The service providing system of claim 7 wherein said first part and second part of said storage system is a program memory.

9. The service providing system of claim 7 wherein said associated login information comprises an anonymous user login mapped to said user identification.

10. The service providing system of claim 7 wherein said user computer system comprises a device selected from a group consisting of a Personal Computer (PC), laptop, Personal Digital Assistant (PDA), cell phone, analog mobile phone, or workstation.

11. The service providing system of claim 7 wherein said service is in the same area as said user.

12. The service providing system of claim 7 wherein said computer system is a local to said user.

13. A distributed system for providing a plurality of services to a plurality of user computer systems, comprising:

a central server for providing a common access point for said plurality of user computer systems to said plurality of services;

a plurality of area servers coupled to said central server for providing information on local services of said plurality of services to local user computer systems of said plurality of user computer systems; and

a plurality of local cooperated company servers having said local services and coupled to said plurality of area servers; and

wherein a local user computer system of said plurality of user computer systems accesses said central server for a user request for a local service, said central server sends said request to a area server of said plurality of area servers, and said area server obtains said local service for said local user computer system.

14. The distributed system of claim 13 wherein said area server has a mapping of a local anonymous user to said local user for access to said local cooperated company server having said local service.

15. A data structure, including a table, stored in a computer readable medium for enabling a user to anonymously receive a service from a second service provider via a first service provider, said table comprising:

a first column entry in a row of said table comprising a user identification for said user, said user identification required for access to said first service provider;

a second column entry in said row of said table, having indentation information for said second service provider; and

a third column entry in said row of said table associated with said user identification and said indentation information for providing a predetermined user code for accessing said service on said second service provider.

16. The data structure of claim 15 wherein said predetermined user code is for an anonymous user.

17. The data structure of claim 15 wherein said predetermined user code is used by said first service provider to logon onto and receive said service from said second service provider on behalf of said user.

18. A computer readable medium for storing code for providing use by a user of a plurality of services through a common access point, comprising:

code for providing said user access to a first computer system;

code for said first computer system accessing a second computer system comprising said service in response to a user request for a service of said plurality of services,; and

code for providing said service to said user from said second computer system through said first computer system.

19. A service providing method for connecting a plurality of service providing systems to a service access system via a network, thereby enabling the plurality of the service providing systems to provide various services to the service access apparatus, the service providing method comprising:

sending display data including an indication for a service request issued to a second service providing system when said user selects said indication, said display data sent, when a service is provided to the service access system from a first service providing system;

enabling the service access apparatus to request a service of the second service providing system by the user selecting said indication from the display data displayed on the screen of the service access apparatus; and

starting communication between the service access apparatus and the second service providing system via the first service providing system, thereby providing the service of the second service providing system to the service access apparatus.

20. The service providing method of claim 19 further comprising:

enabling the service access apparatus to log in to said first service providing system with use of a predetermined user ID, thereby establishing communication between the service access apparatus and the first service providing system.

21. The service providing method of claim 19, further comprising the first service providing system logging into the second service providing system with a preset anonymous user ID before said starting communication between

the service access apparatus and the second service providing system via the first service providing system.

22. A service providing system for providing various services to a service access apparatus via a network, the system comprising:

a means for sending display data including a selection button for a service request to a second service providing system, when a service is provided to the service access apparatus;

a means for establishing communication with the second service providing system when the selection button displayed on the screen of the service access apparatus is used; and

a means for communicating between the second service providing system and the service access apparatus so as to enable the second service providing system to provide the service to the service access apparatus via the first service providing system.

* * * * *

US 20020059080A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2002/0059080 A1

Kasirer et al. (43) Pub. Date: May 16, 2002

(54) SYSTEM, METHOD, AND USER INTERFACE FOR MANAGING INTERMEDIATE HEALTHCARE FACILITIES OVER COMPUTER NETWORKS

(76) Inventors: **Robert Kasirer**, Beverly Hills, CA (US); **Florian Wieland**, Laguna Hills, CA (US); **James Koontz**, Diamond Bar, CA (US); **Steve Mayner**, Marina del Rey, CA (US); **Carl Strunk**, Rancho Palos Verdes, CA (US)

Correspondence Address:
MCDERMOTT, WILL & EMERY
600 13th Street, N.W.
Washington, DC 20005-3096 (US)

(21) Appl. No.: **09/942,942**

(22) Filed: **Aug. 31, 2001**

Related U.S. Application Data

(63) Non-provisional of provisional application No. 60/230,218, filed on Sep. 1, 2000. Non-provisional of provisional application No. 60/265,186, filed on Jan. 30, 2001. Non-provisional of provisional application No. 60/282,876, filed on Apr. 11, 2001.

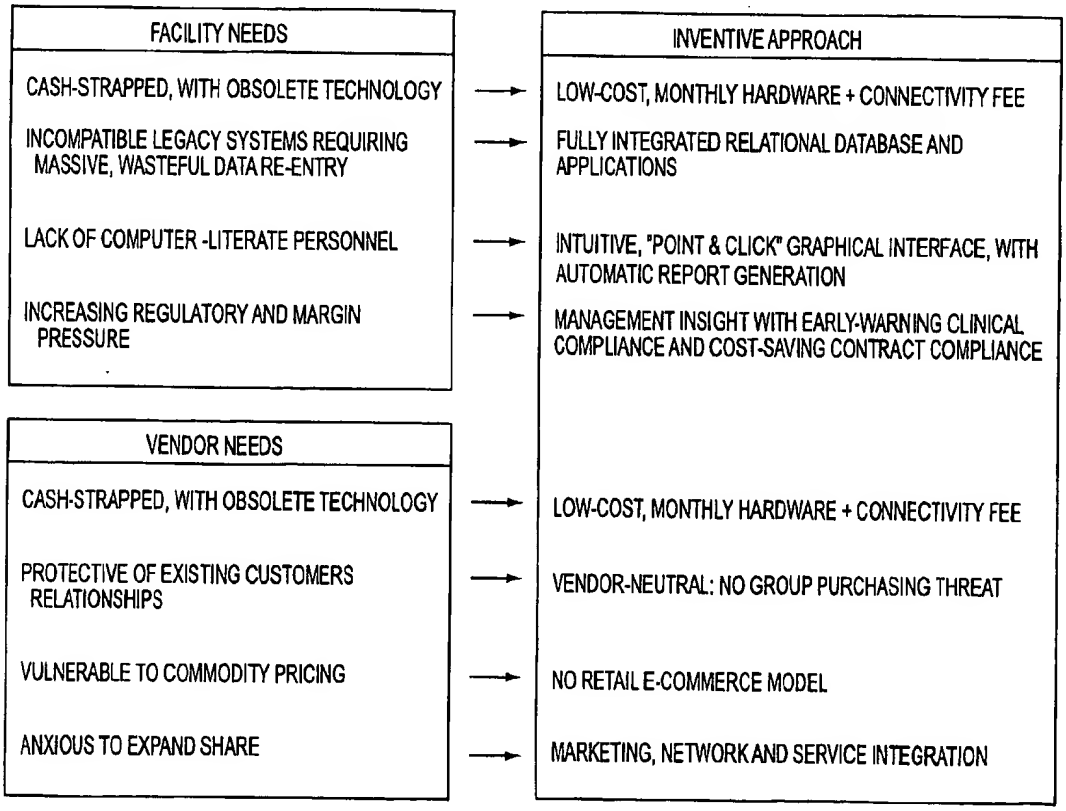
Publication Classification

(51) **Int. Cl.**[7] **G06F 17/60**

(52) **U.S. Cl.** **705/2**

(57) **ABSTRACT**

A variety of techniques are directed to providing services and technology to Intermediate Care Facilities, such as small rural hospitals, psychiatric institutions, nursing homes and assisted living facilities. An integrated suite of applications tailored to the needs of such Facilities is provided over a network from an application service provider. A central server provides links to vendors and ensures an integrated procurement process that enforces the rules of the particular facility.



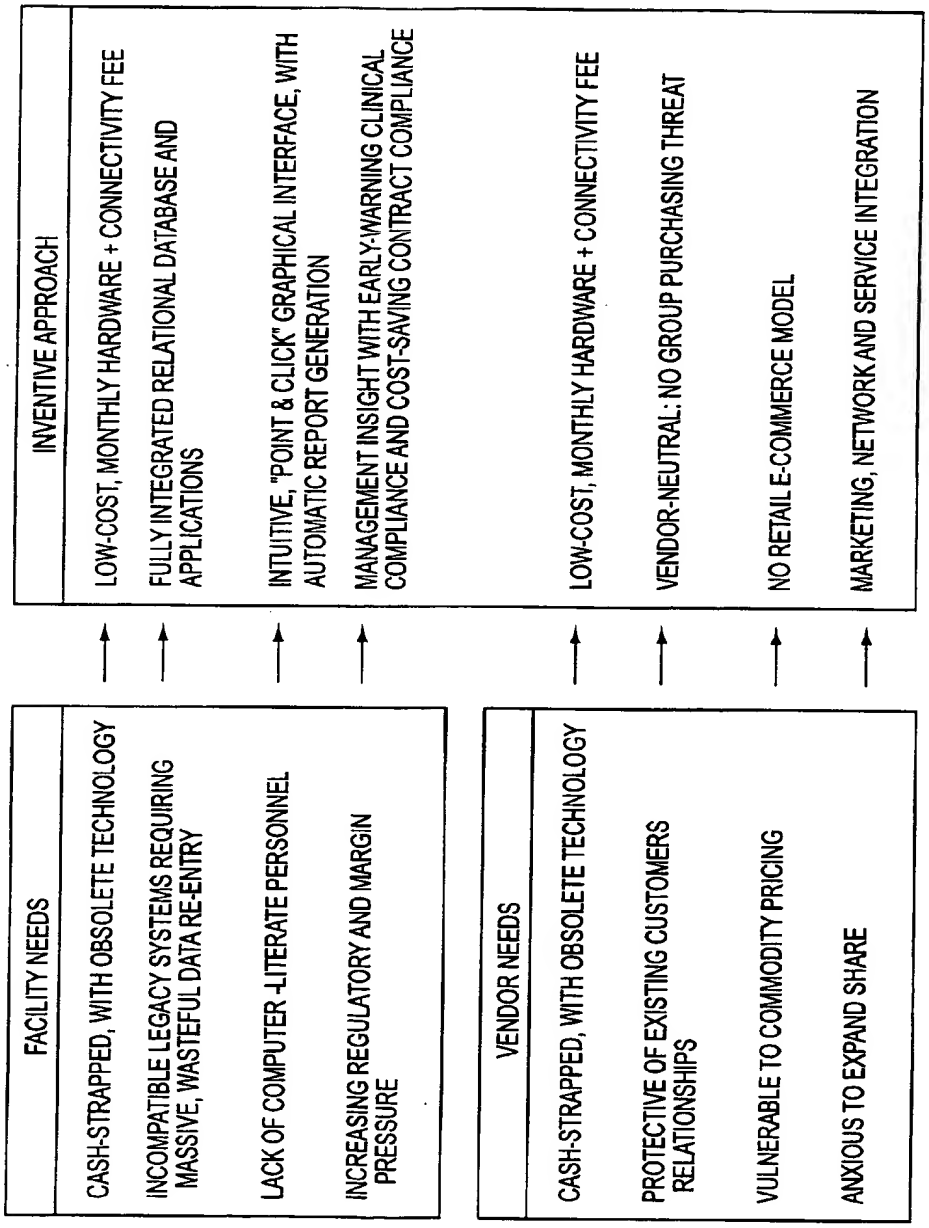

FACILITY NEEDS
CASH-STRAPPED, WITH OBSOLETE TECHNOLOGY
INCOMPATIBLE LEGACY SYSTEMS REQUIRING MASSIVE, WASTEFUL DATA RE-ENTRY
LACK OF COMPUTER -LITERATE PERSONNEL
INCREASING REGULATORY AND MARGIN PRESSURE
VENDOR NEEDS
CASH-STRAPPED, WITH OBSOLETE TECHNOLOGY
PROTECTIVE OF EXISTING CUSTOMERS RELATIONSHIPS
VULNERABLE TO COMMODITY PRICING
ANXIOUS TO EXPAND SHARE
INVENTIVE APPROACH
LOW-COST, MONTHLY HARDWARE + CONNECTIVITY FEE
FULLY INTEGRATED RELATIONAL DATABASE AND APPLICATIONS
INTUITIVE, "POINT & CLICK" GRAPHICAL INTERFACE, WITH AUTOMATIC REPORT GENERATION
MANAGEMENT INSIGHT WITH EARLY-WARNING CLINICAL COMPLIANCE AND COST-SAVING CONTRACT COMPLIANCE
LOW-COST, MONTHLY HARDWARE + CONNECTIVITY FEE
VENDOR-NEUTRAL: NO GROUP PURCHASING THREAT
NO RETAIL E-COMMERCE MODEL
MARKETING, NETWORK AND SERVICE INTEGRATION

FIG. 1

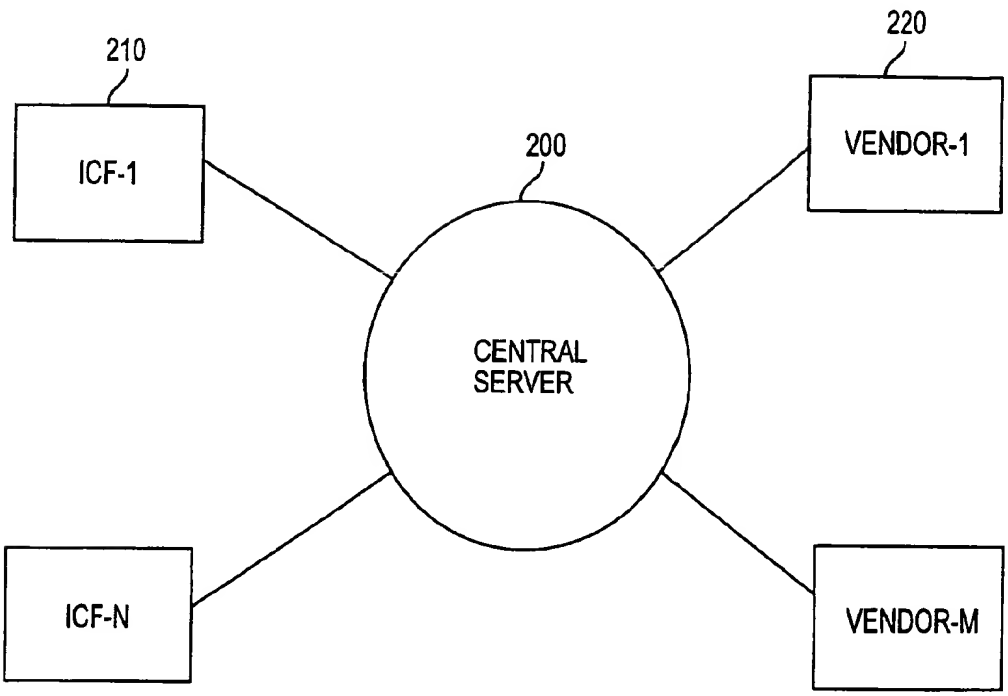


FIG. 2

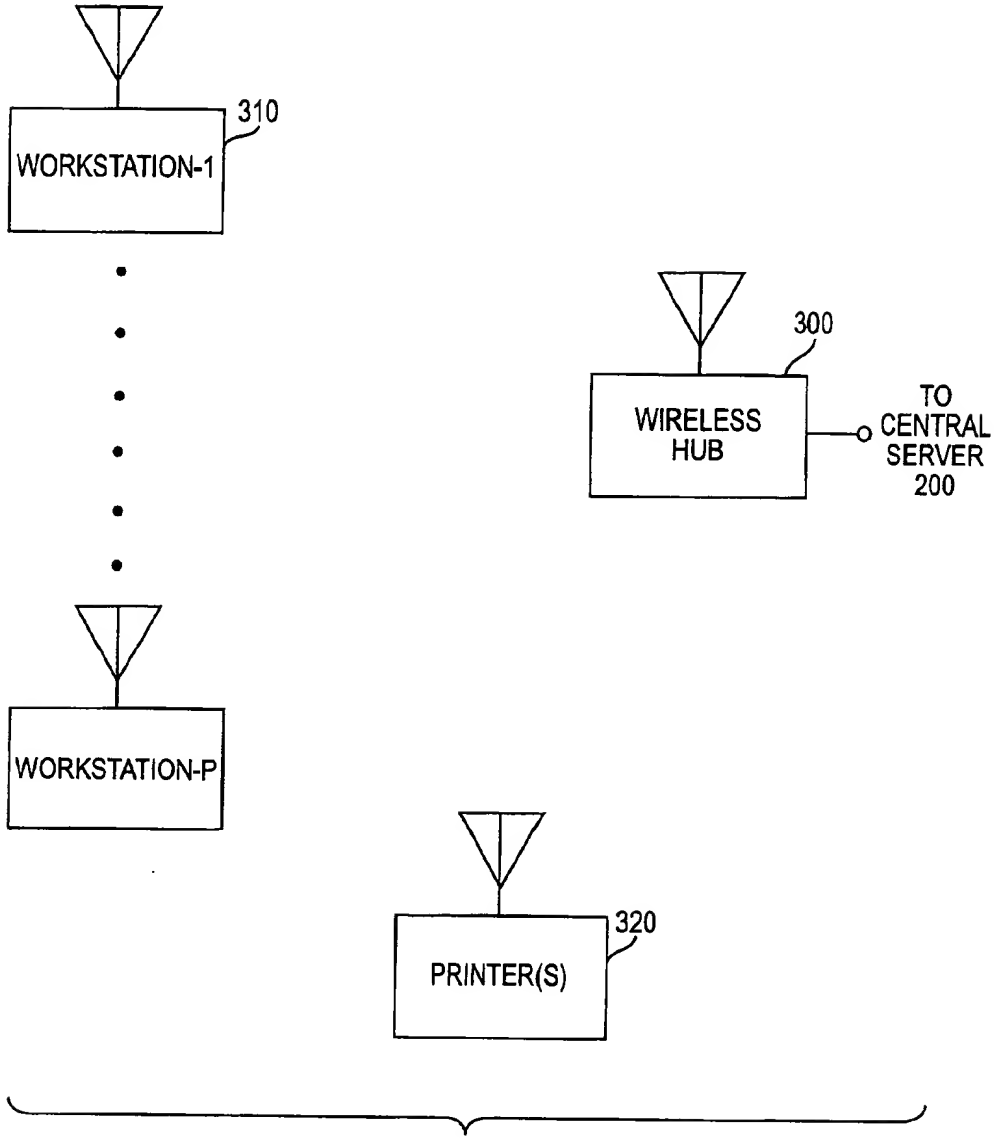

310
WORKSTATION-1
300
WIRELESS HUB
TO CENTRAL SERVER 200
WORKSTATION-P
320
PRINTER(S)

FIG. 3

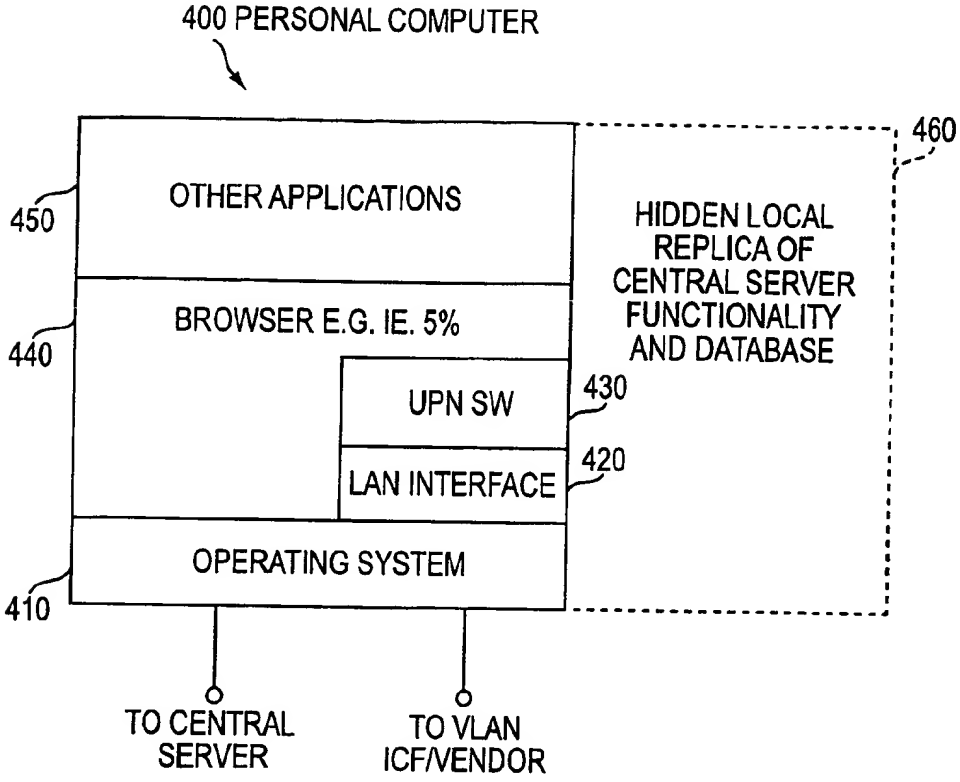

400 PERSONAL COMPUTER
460
OTHER APPLICATIONS
450
HIDDEN LOCAL REPLICA OF CENTRAL SERVER FUNCTIONALITY AND DATABASE
BROWSER E.G. IE. 5%
440
UPN SW
430
LAN INTERFACE
420
OPERATING SYSTEM
410
TO CENTRAL SERVER
TO VLAN ICF/VENDOR

FIG. 4

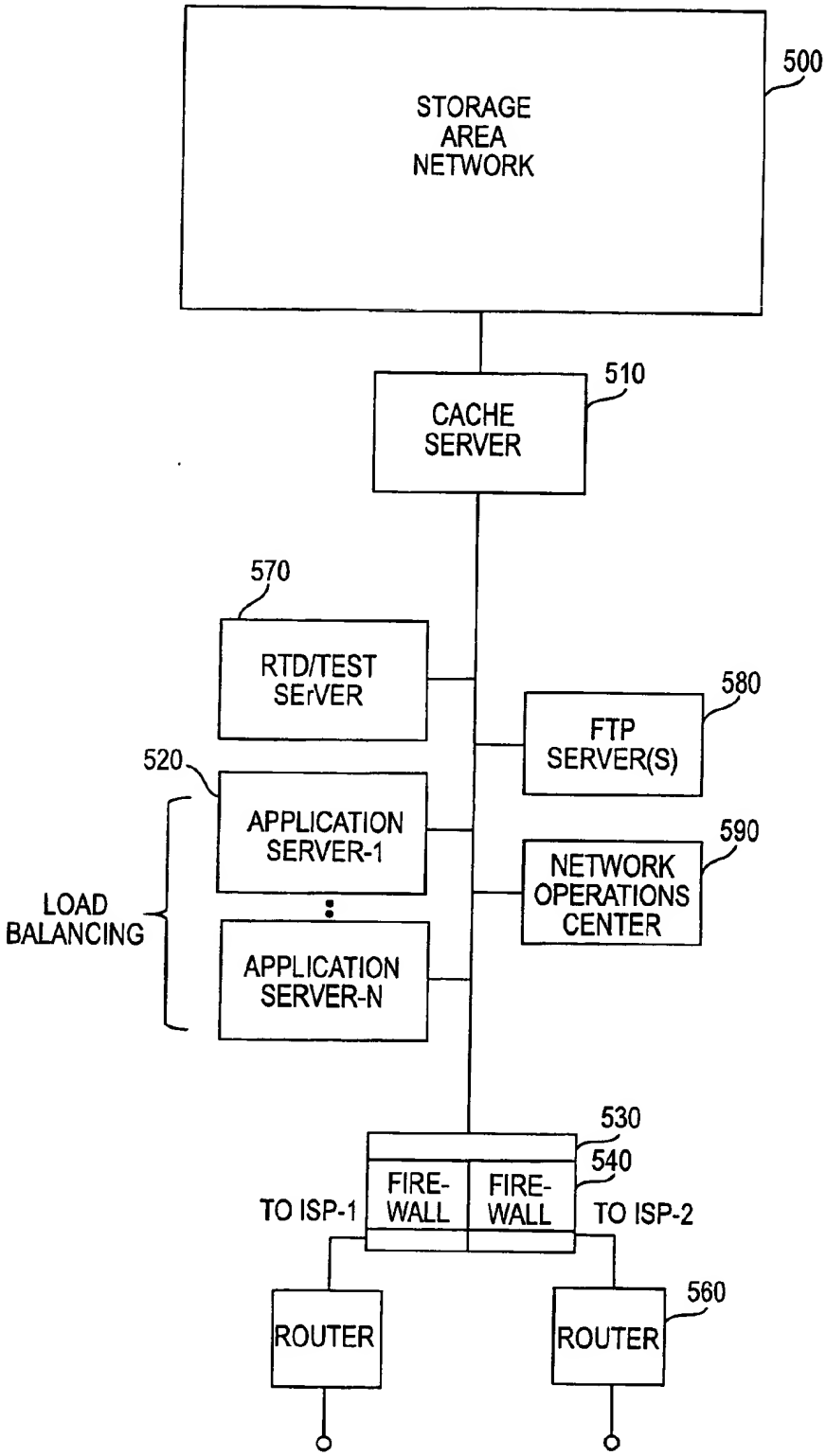

500
STORAGE AREA NETWORK
510
CACHE SERVER
570
RTD/TEST SERVER
580
FTP SERVER(S)
520
APPLICATION SERVER-1
590
NETWORK OPERATIONS CENTER
LOAD BALANCING
APPLICATION SERVER-N
530
540
FIRE-WALL
FIRE-WALL
TO ISP-1
TO ISP-2
560
ROUTER
ROUTER

FIG. 5

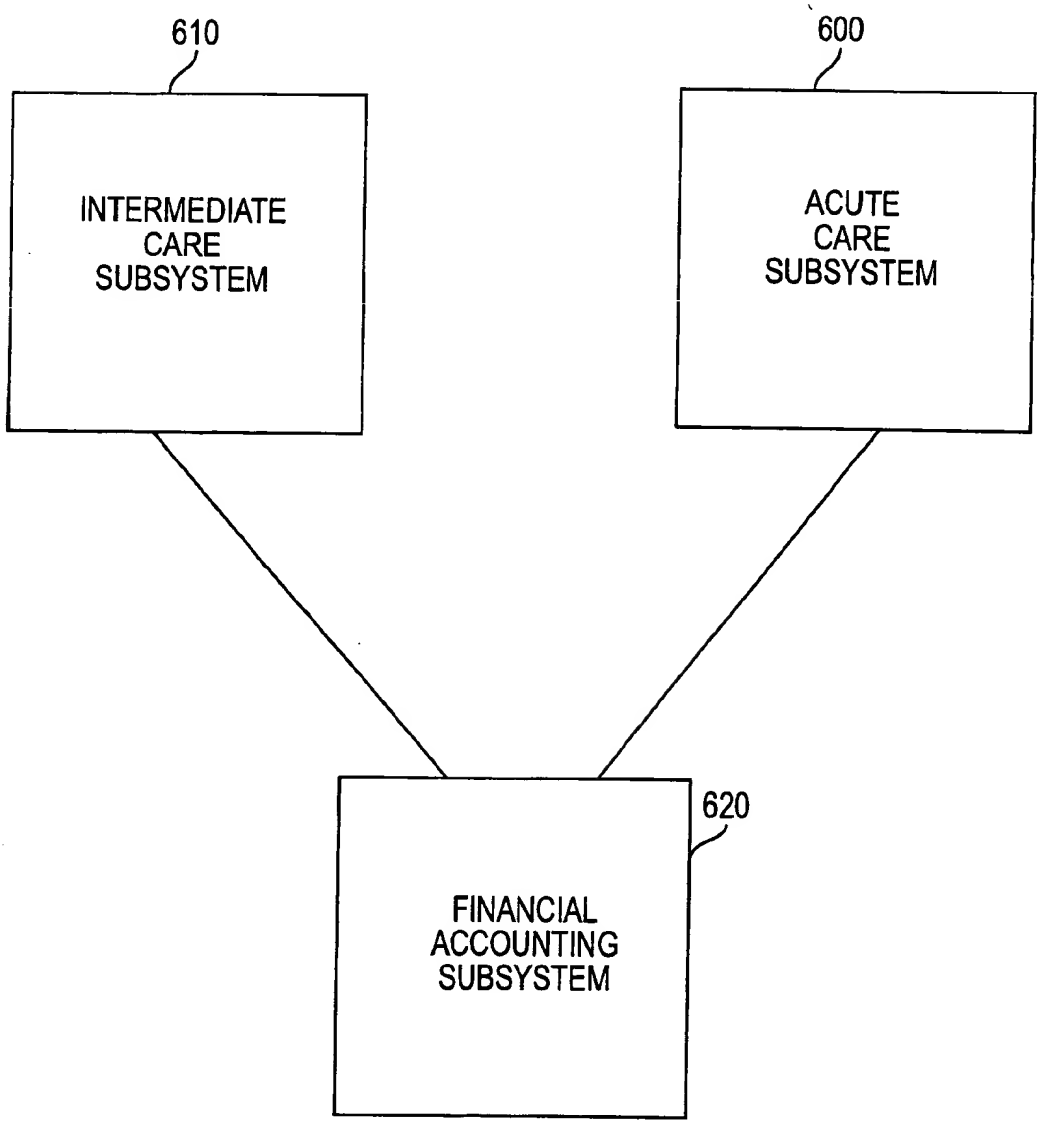

610
600
INTERMEDIATE CARE SUBSYSTEM
ACUTE CARE SUBSYSTEM
620
FINANCIAL ACCOUNTING SUBSYSTEM

FIG. 6

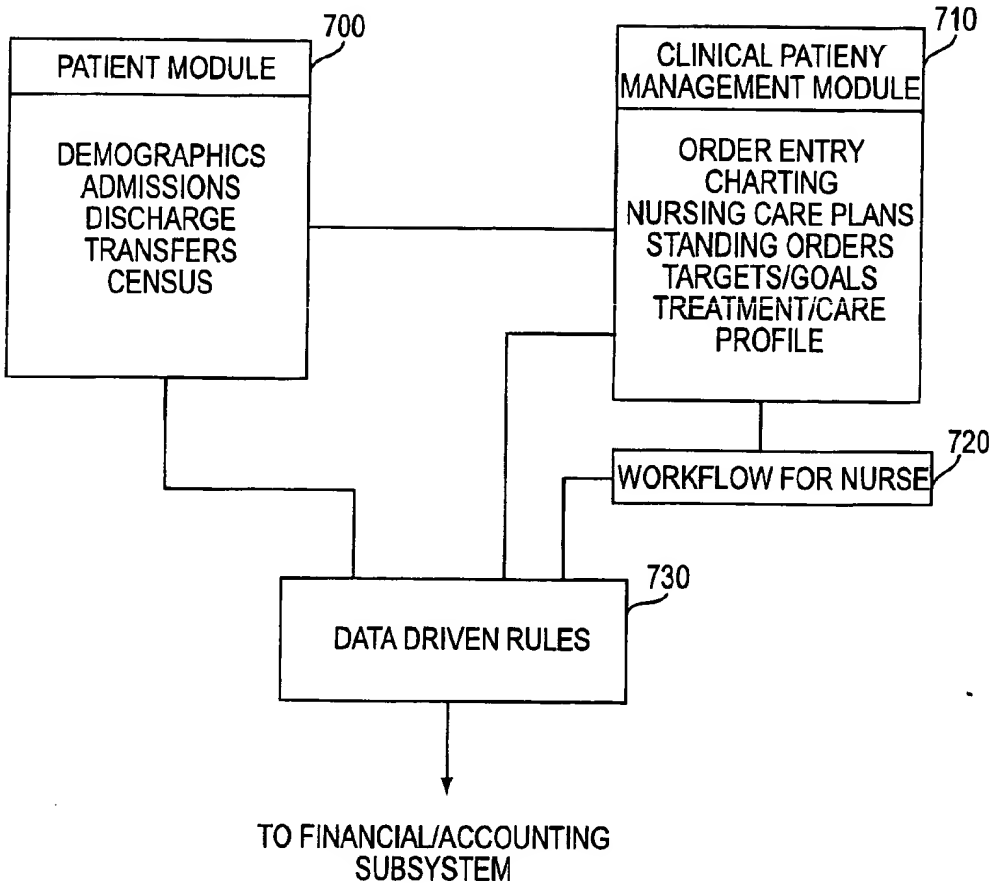

700
PATIENT MODULE
DEMOGRAPHICS
ADMISSIONS
DISCHARGE
TRANSFERS
CENSUS
710
CLINICAL PATIENY MANAGEMENT MODULE
ORDER ENTRY
CHARTING
NURSING CARE PLANS
STANDING ORDERS
TARGETS/GOALS
TREATMENT/CARE PROFILE
720
WORKFLOW FOR NURSE
730
DATA DRIVEN RULES
TO FINANCIAL/ACCOUNTING SUBSYSTEM

FIG. 7

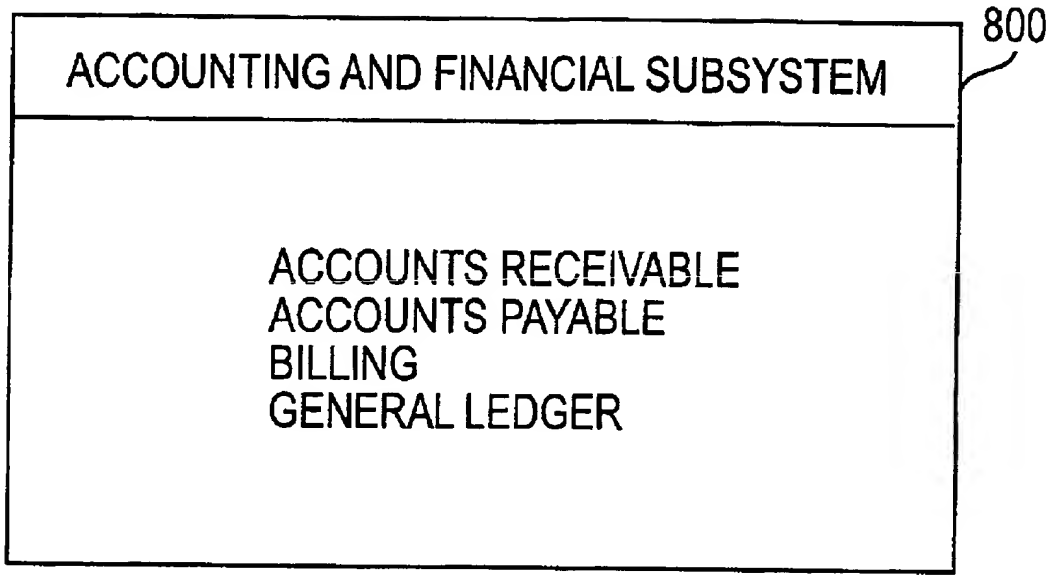

800
ACCOUNTING AND FINANCIAL SUBSYSTEM
ACCOUNTS RECEIVABLE
ACCOUNTS PAYABLE
BILLING
GENERAL LEDGER

FIG. 8

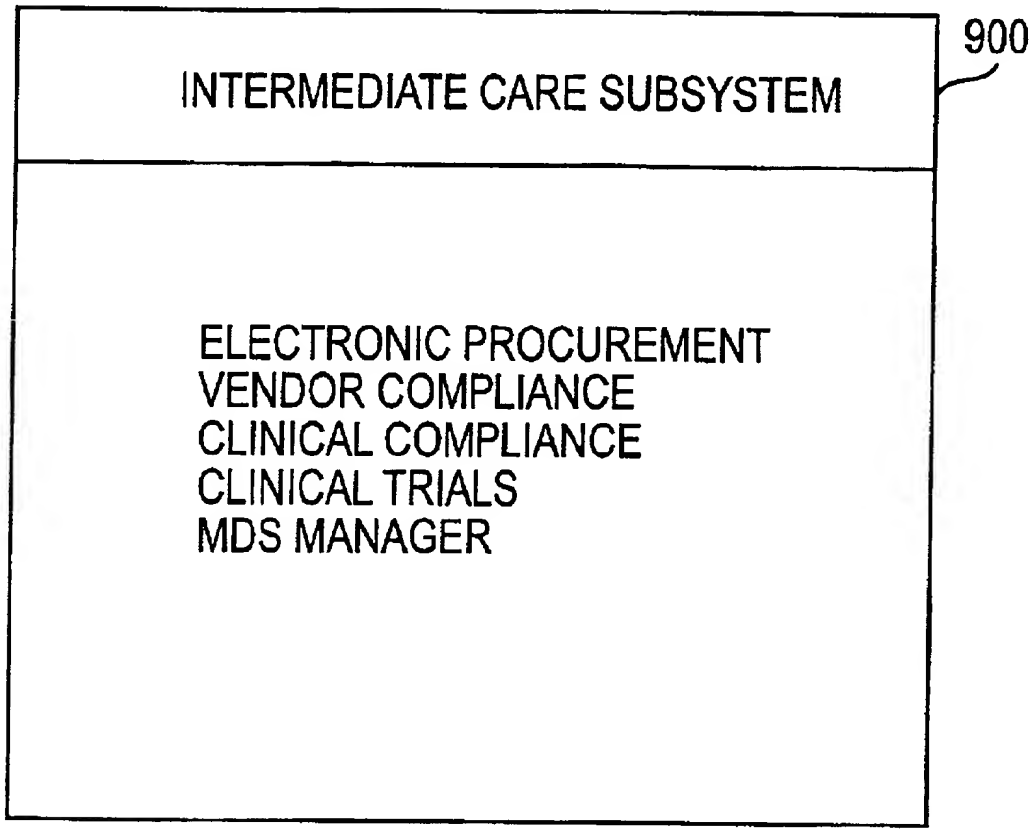

900
INTERMEDIATE CARE SUBSYSTEM
ELECTRONIC PROCUREMENT
VENDOR COMPLIANCE
CLINICAL COMPLIANCE
CLINICAL TRIALS
MDS MANAGER

FIG. 9

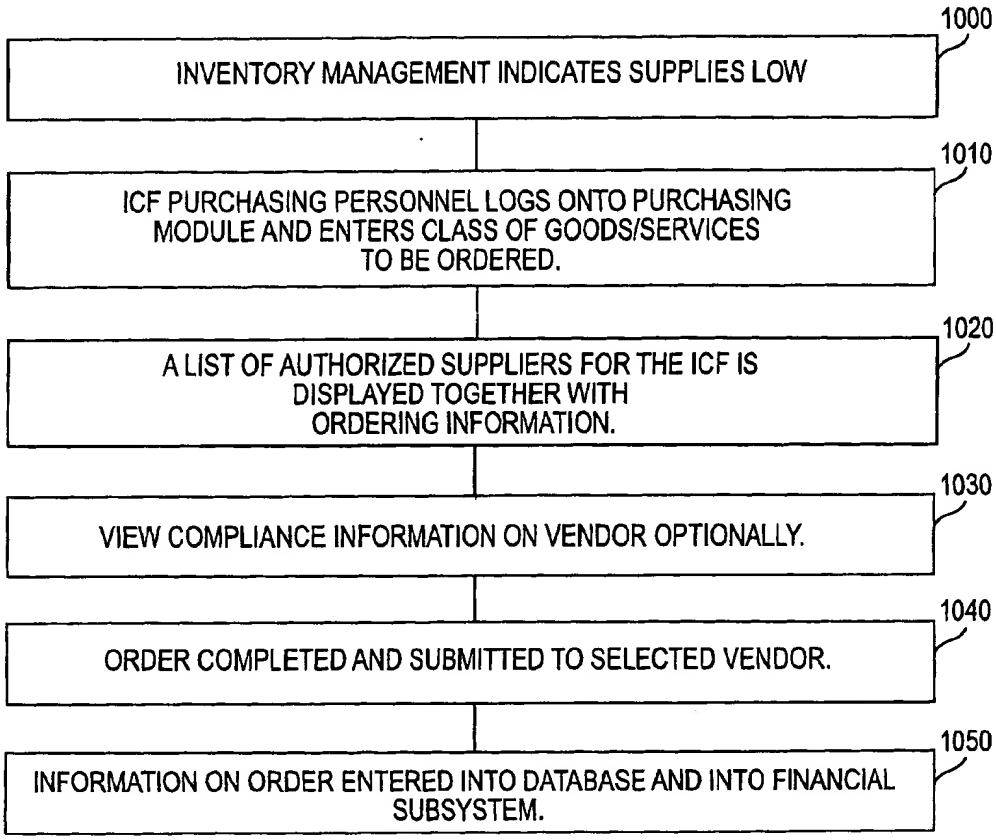

1000
INVENTORY MANAGEMENT INDICATES SUPPLIES LOW
1010
ICF PURCHASING PERSONNEL LOGS ONTO PURCHASING MODULE AND ENTERS CLASS OF GOODS/SERVICES TO BE ORDERED.
1020
A LIST OF AUTHORIZED SUPPLIERS FOR THE ICF IS DISPLAYED TOGETHER WITH ORDERING INFORMATION.
1030
VIEW COMPLIANCE INFORMATION ON VENDOR OPTIONALLY.
1040
ORDER COMPLETED AND SUBMITTED TO SELECTED VENDOR.
1050
INFORMATION ON ORDER ENTERED INTO DATABASE AND INTO FINANCIAL SUBSYSTEM.

FIG. 10

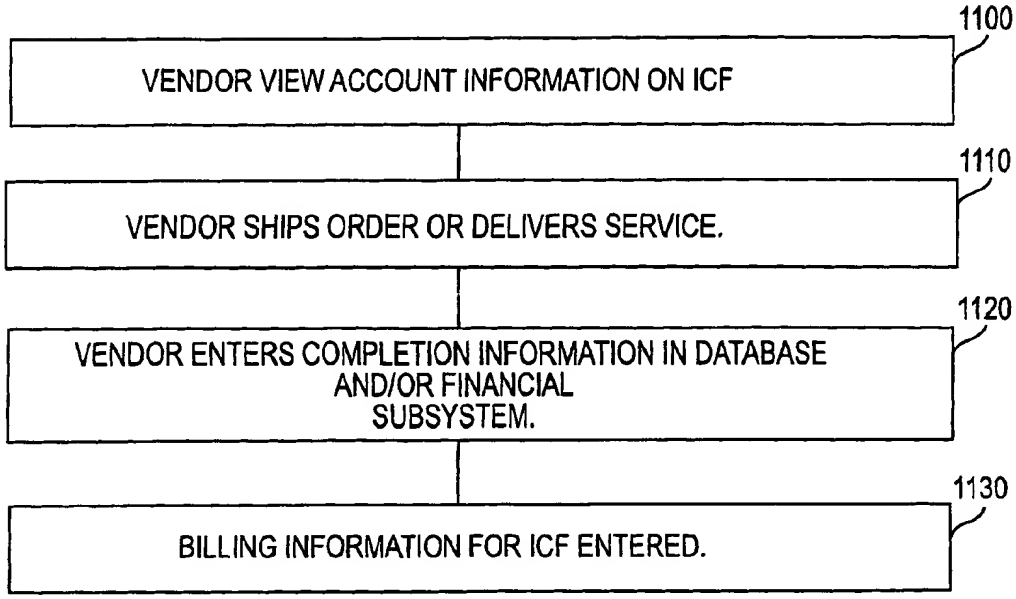

1100
VENDOR VIEW ACCOUNT INFORMATION ON ICF
1110
VENDOR SHIPS ORDER OR DELIVERS SERVICE.
1120
VENDOR ENTERS COMPLETION INFORMATION IN DATABASE AND/OR FINANCIAL SUBSYSTEM.
1130
BILLING INFORMATION FOR ICF ENTERED.

FIG. 11

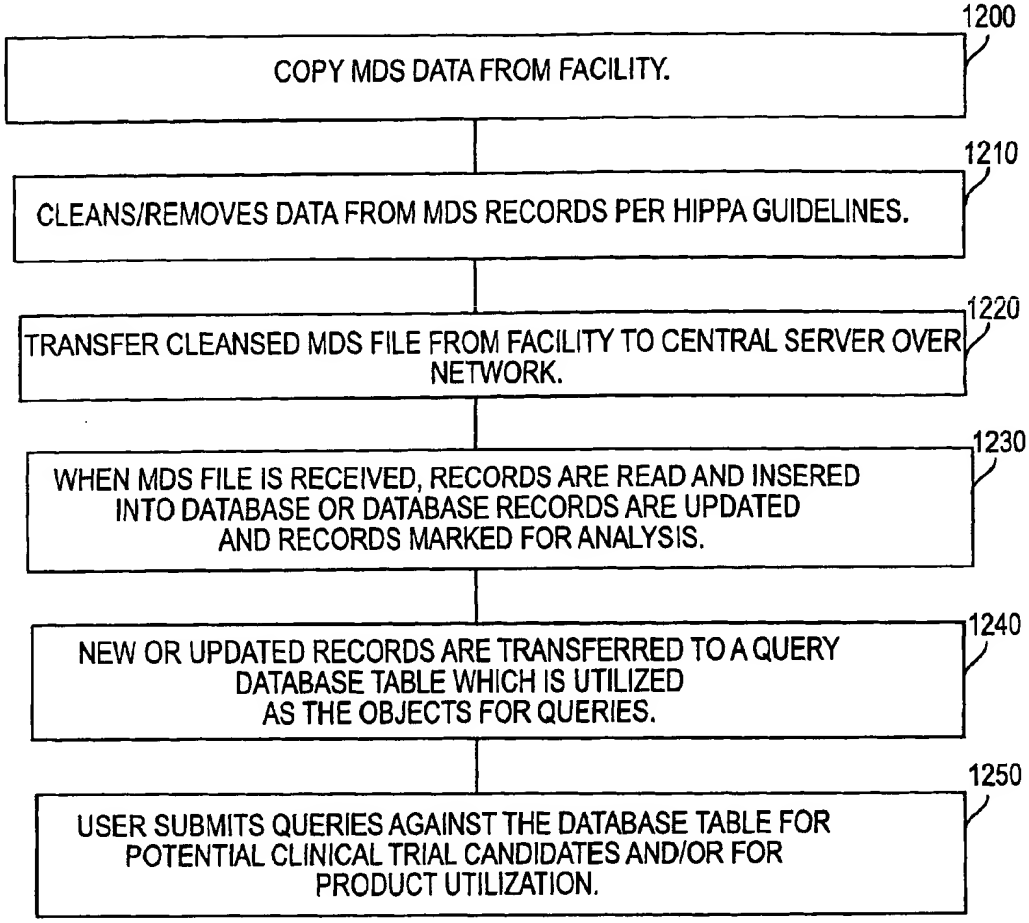

1200
COPY MDS DATA FROM FACILITY.
1210
CLEANS/REMOVES DATA FROM MDS RECORDS PER HIPPA GUIDELINES.
1220
TRANSFER CLEANSED MDS FILE FROM FACILITY TO CENTRAL SERVER OVER NETWORK.
1230
WHEN MDS FILE IS RECEIVED, RECORDS ARE READ AND INSERED INTO DATABASE OR DATABASE RECORDS ARE UPDATED AND RECORDS MARKED FOR ANALYSIS.
1240
NEW OR UPDATED RECORDS ARE TRANSFERRED TO A QUERY DATABASE TABLE WHICH IS UTILIZED AS THE OBJECTS FOR QUERIES.
1250
USER SUBMITS QUERIES AGAINST THE DATABASE TABLE FOR POTENTIAL CLINICAL TRIAL CANDIDATES AND/OR FOR PRODUCT UTILIZATION.

FIG. 12

I. SYSTEM WIDE

A. NORTH AMERICA (ENTERPRISE)

1) CANADA (SUB ENTERPRISE)

A) ONTARIO

B) BRITISH COLUMBIA

2) UNITED STATES OF AMERICA (SUB ENTERPRISE)

A) GLOBAL HEALTH

B) COLUMBIA HEALTH

(1) WEST COAST

(A) CALIFORNIA

— HOSPITAL 1 (FACILITY)

— LONGTERM CARE FACILITY 16 (FACILITY)

— PSYCHIATRIC HOSPITAL 2

— CLINIC 3 (INSTITUTION)

— LONG TERM CARE UNIT

— CLINIC 4

(2) EAST COAST

(3) MIDWEST

FIG. 13

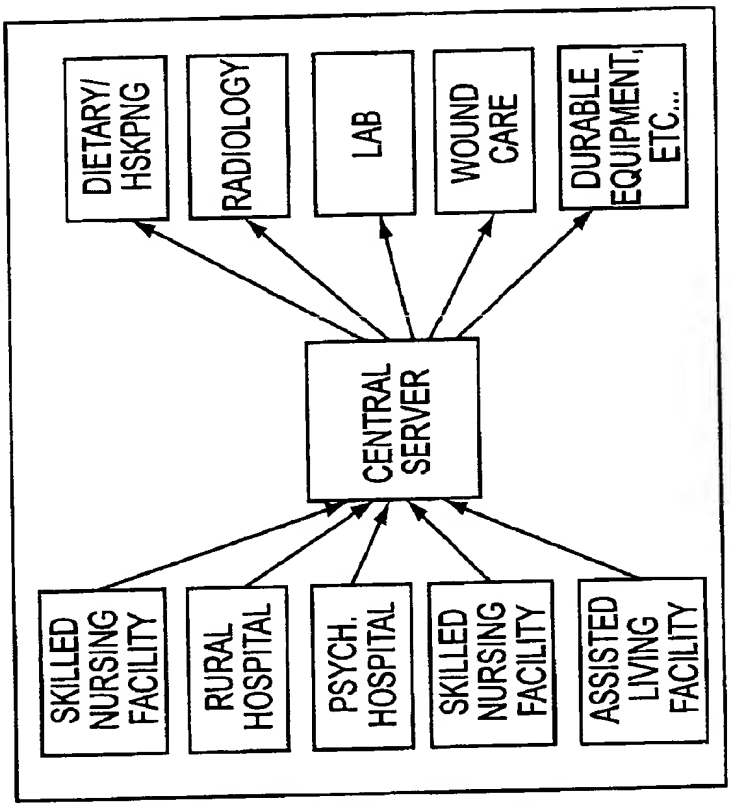


FIG. 14B

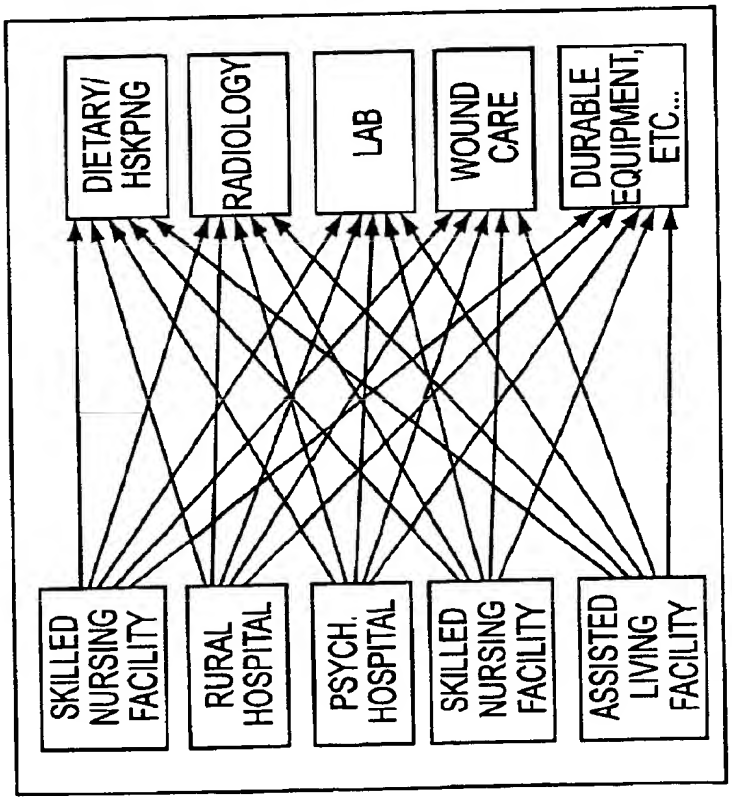


FIG. 14A

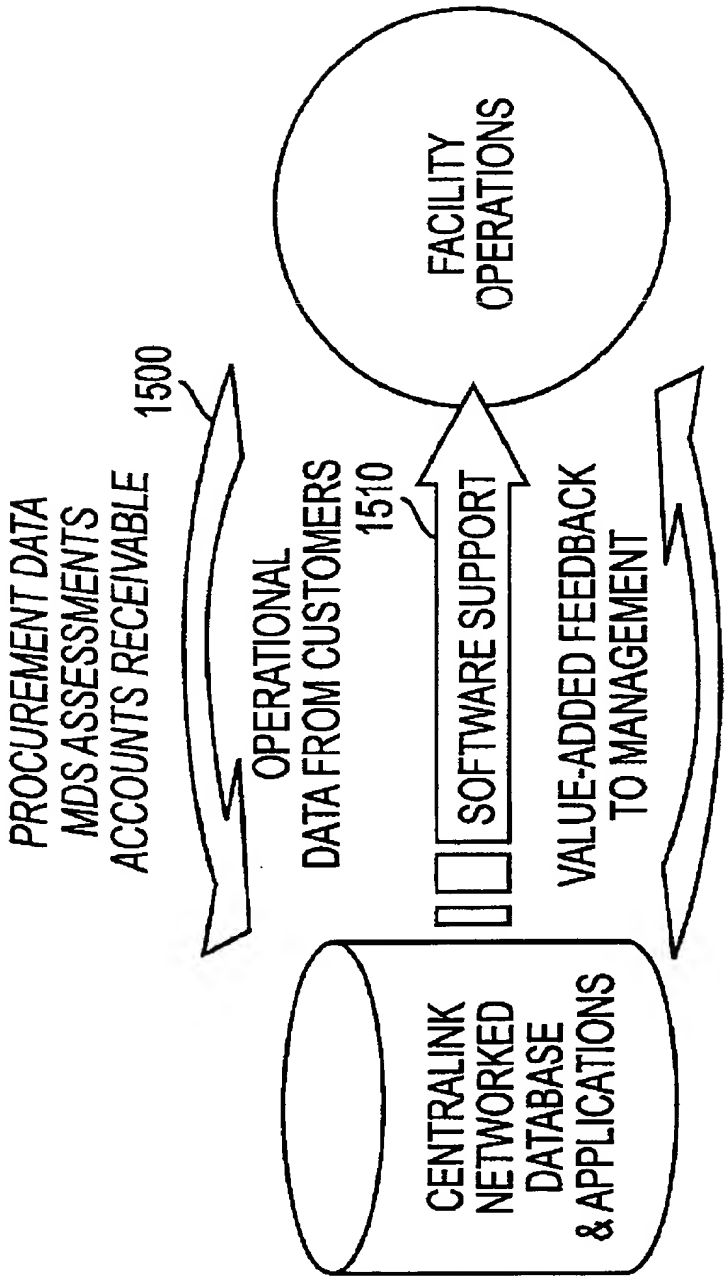

FACILITY OPERATIONS
1500
PROCUREMENT DATA
MDS ASSESSMENTS
ACCOUNTS RECEIVABLE
OPERATIONAL
DATA FROM CUSTOMERS
1510
SOFTWARE SUPPORT
VALUE-ADDED FEEDBACK
TO MANAGEMENT
CENTRALINK
NETWORKED
DATABASE
& APPLICATIONS

FIG. 15

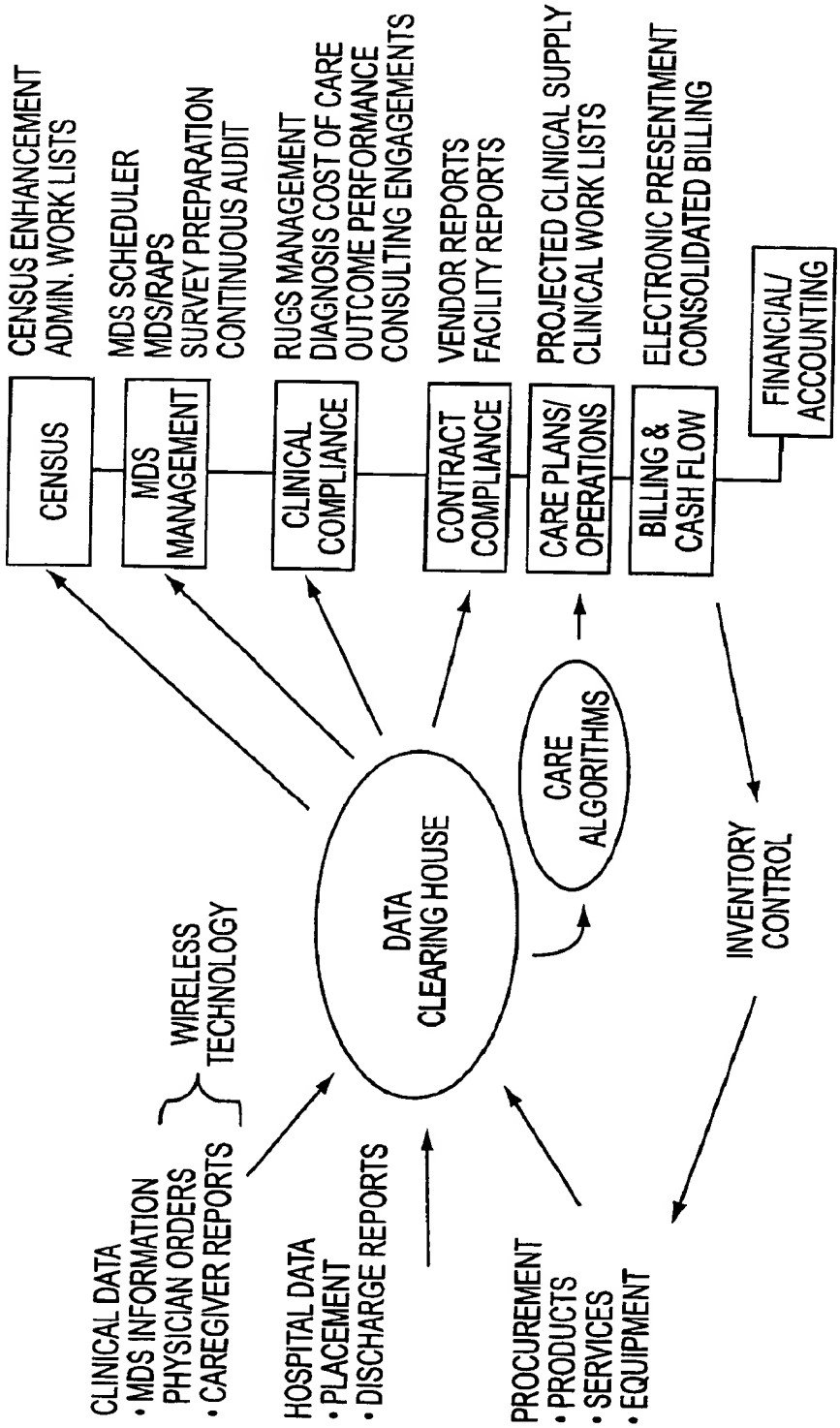

CENSUS ENHANCEMENT
ADMIN. WORK LISTS
MDS SCHEDULER
MDS/RAPS
SURVEY PREPARATION
CONTINUOUS AUDIT
RUGS MANAGEMENT
DIAGNOSIS COST OF CARE
OUTCOME PERFORMANCE
CONSULTING ENGAGEMENTS
VENDOR REPORTS
FACILITY REPORTS
PROJECTED CLINICAL SUPPLY
CLINICAL WORK LISTS
ELECTRONIC PRESENTMENT
CONSOLIDATED BILLING
CENSUS
MDS MANAGEMENT
CLINICAL COMPLIANCE
CONTRACT COMPLIANCE
CARE PLANS/ OPERATIONS
BILLING & CASH FLOW
FINANCIAL/ ACCOUNTING
CARE ALGORITHMS
DATA CLEARING HOUSE
INVENTORY CONTROL
WIRELESS TECHNOLOGY
CLINICAL DATA
• MDS INFORMATION
PHYSICIAN ORDERS
• CAREGIVER REPORTS
HOSPITAL DATA
• PLACEMENT
• DISCHARGE REPORTS
PROCUREMENT
• PRODUCTS
• SERVICES
• EQUIPMENT

FIG. 16

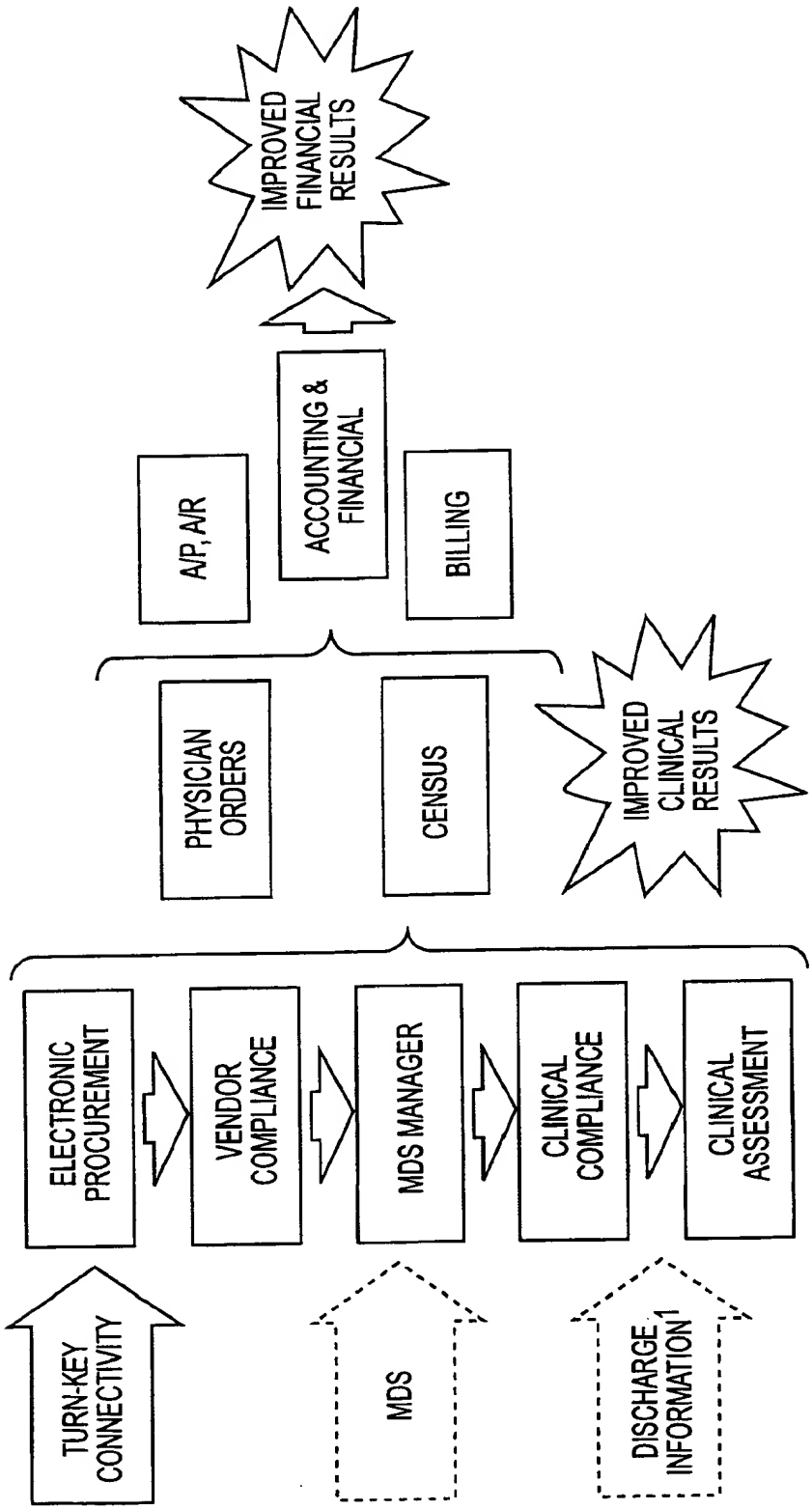

IMPROVED FINANCIAL RESULTS
A/P, A/R
ACCOUNTING & FINANCIAL
BILLING
PHYSICIAN ORDERS
CENSUS
IMPROVED CLINICAL RESULTS
ELECTRONIC PROCUREMENT
VENDOR COMPLIANCE
MDS MANAGER
CLINICAL COMPLIANCE
CLINICAL ASSESSMENT
TURN-KEY CONNECTIVITY
MDS
DISCHARGE INFORMATION[1]

FIG. 17

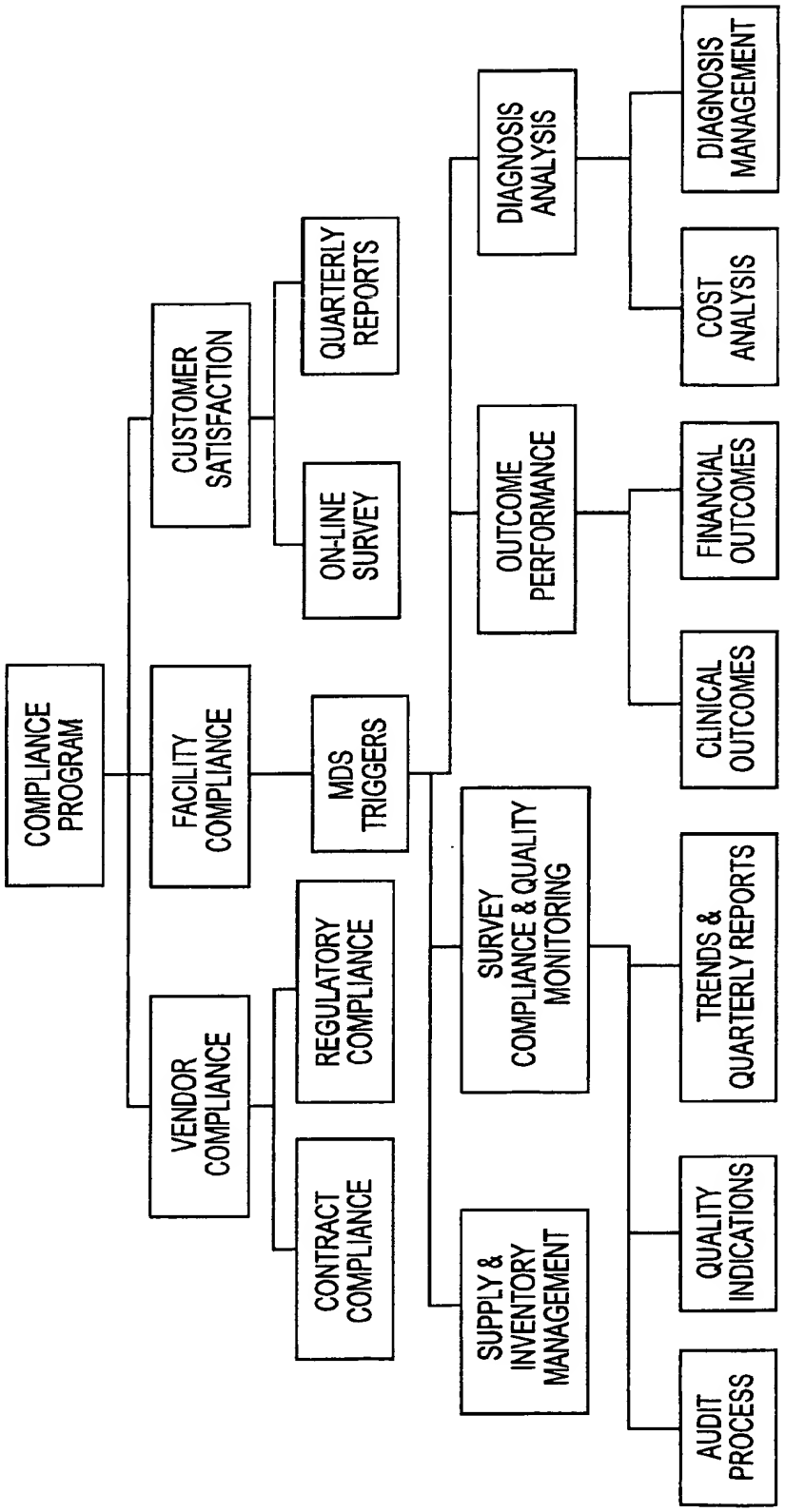

COMPLIANCE PROGRAM
FACILITY COMPLIANCE
VENDOR COMPLIANCE
CUSTOMER SATISFACTION
ON-LINE SURVEY
QUARTERLY REPORTS
MDS TRIGGERS
REGULATORY COMPLIANCE
CONTRACT COMPLIANCE
SURVEY COMPLIANCE & QUALITY MONITORING
SUPPLY & INVENTORY MANAGEMENT
OUTCOME PERFORMANCE
DIAGNOSIS ANALYSIS
DIAGNOSIS MANAGEMENT
COST ANALYSIS
FINANCIAL OUTCOMES
CLINICAL OUTCOMES
TRENDS & QUARTERLY REPORTS
QUALITY INDICATIONS
AUDIT PROCESS

FIG. 18

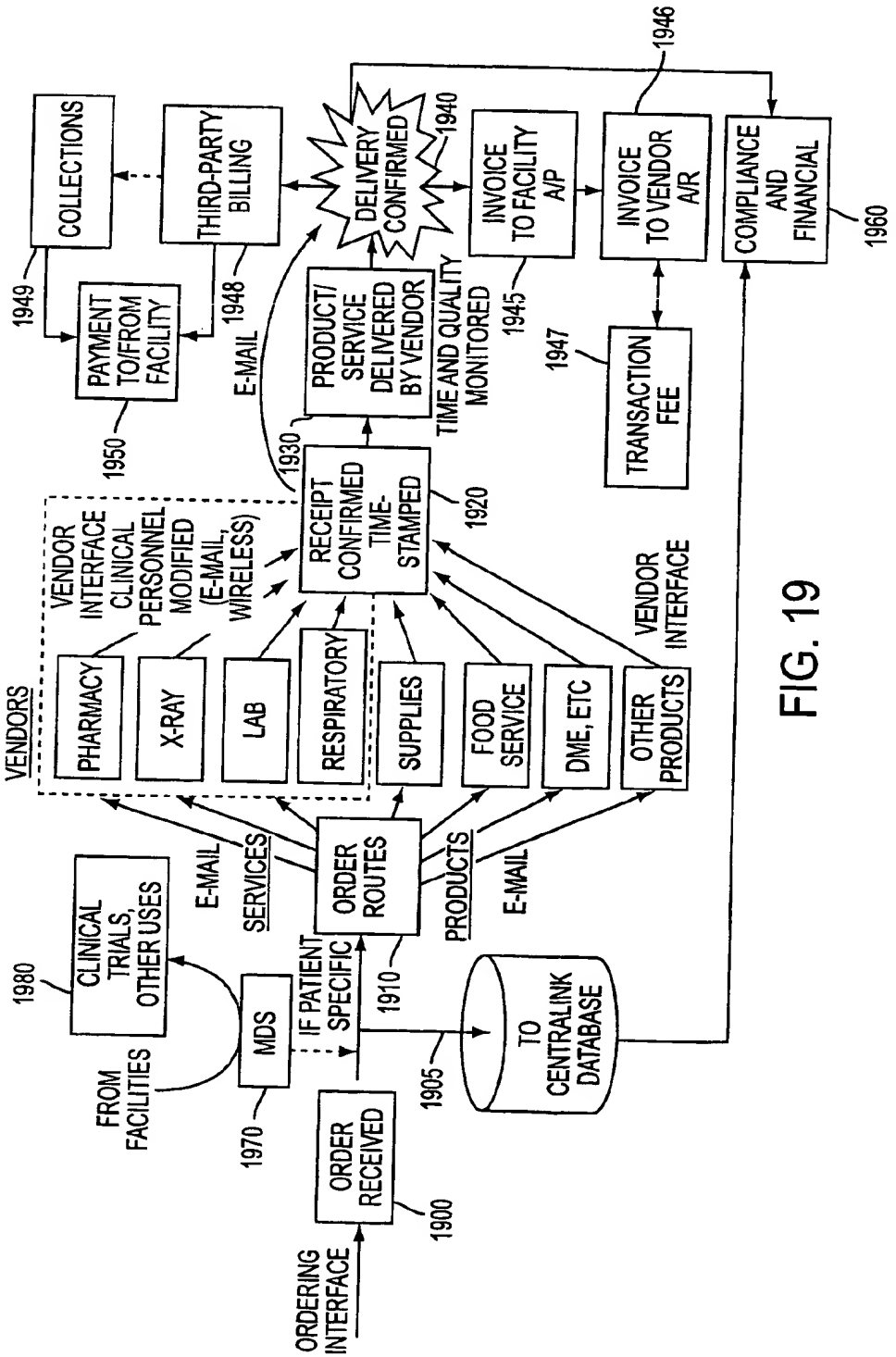
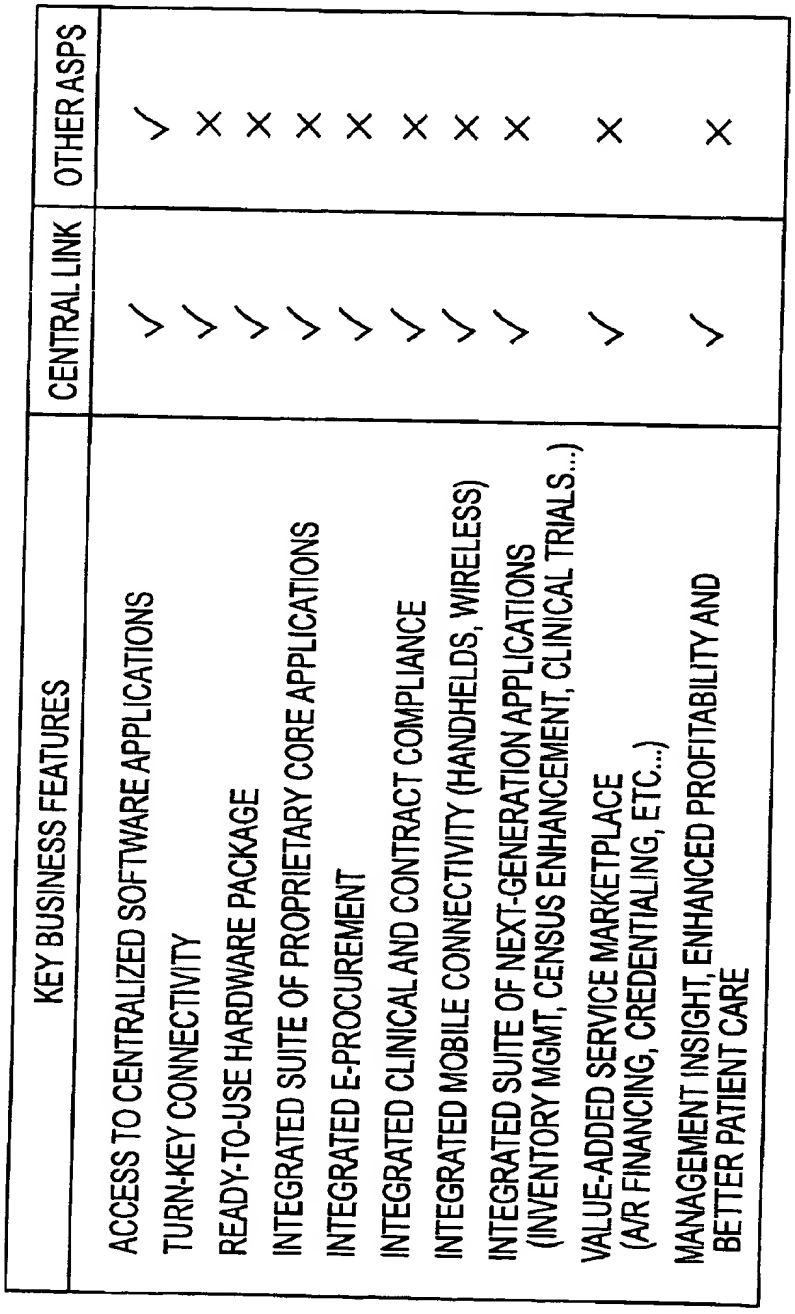

ORDERING INTERFACE
ORDER RECEIVED
1900
FROM FACILITIES
MDS
1970
IF PATIENT SPECIFIC
CLINICAL TRIALS, OTHER USES
1980
E-MAIL
SERVICES
ORDER ROUTES
1910
PRODUCTS
E-MAIL
TO CENTRALINK DATABASE
1905
VENDORS
PHARMACY
X-RAY
LAB
RESPIRATORY
SUPPLIES
FOOD SERVICE
DME, ETC
OTHER PRODUCTS
VENDOR INTERFACE
VENDOR INTERFACE CLINICAL PERSONNEL MODIFIED (E-MAIL, WIRELESS)
RECEIPT CONFIRMED TIME-STAMPED
1920
1930
E-MAIL
PRODUCT/ SERVICE DELIVERED BY VENDOR
TIME AND QUALITY MONITORED
DELIVERY CONFIRMED
1940
1946
INVOICE TO FACILITY A/P
1945
INVOICE TO VENDOR A/R
COMPLIANCE AND FINANCIAL
1960
1947
TRANSACTION FEE
COLLECTIONS
1949
THIRD-PARTY BILLING
1948
PAYMENT TO/FROM FACILITY
1950

FIG. 19



KEY BUSINESS FEATURES	CENTRAL LINK	OTHER ASPS
ACCESS TO CENTRALIZED SOFTWARE APPLICATIONS	✓	✓
TURN-KEY CONNECTIVITY	✓	X
READY-TO-USE HARDWARE PACKAGE	✓	X
INTEGRATED SUITE OF PROPRIETARY CORE APPLICATIONS	✓	X
INTEGRATED E-PROCUREMENT	✓	X
INTEGRATED CLINICAL AND CONTRACT COMPLIANCE	✓	X
INTEGRATED MOBILE CONNECTIVITY (HANDHELDS, WIRELESS)	✓	X
INTEGRATED SUITE OF NEXT-GENERATION APPLICATIONS (INVENTORY MGMT, CENSUS ENHANCEMENT, CLINICAL TRIALS...)	✓	X
VALUE-ADDED SERVICE MARKETPLACE (A/R FINANCING, CREDENTIALING, ETC...)	✓	X
MANAGEMENT INSIGHT, ENHANCED PROFITABILITY AND BETTER PATIENT CARE	✓	X

FIG. 20

SYSTEM, METHOD, AND USER INTERFACE FOR MANAGING INTERMEDIATE HEALTHCARE FACILITIES OVER COMPUTER NETWORKS

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority from and is related to U.S. Provisional Application No. 60/230,218, filed Sep. 1, 2000, Attorney Docket Number 57177-013, and to U.S. Provisional Application No. 60/265,186, filed Jan. 30, 2001, with Attorney Docket Number 57177-016 and to U.S. Provisional Application No. 60/282,876, filed Apr. 11, 2001, with Attorney Docket Number 57177-017. The contents of each of those provisional applications are hereby incorporated by reference in their entirety.

CROSS REFERENCE TO CD-ROM APPENDICES

[0002] This application contains an Appendix containing three (3) CD-ROMs, each containing instructions for implementations of various portions of the computer programs used to carry out the invention disclosed herein. The contents of the CD-ROMs are described in more detail in paper Appendix A attached to this document.

[0003] 1. Field of the Invention

[0004] The present invention relates to managing Intermediate Care Facilities (ICF). More particularly, the present invention provides techniques, systems, methods, and user interfaces for managing such Intermediate Care Facilities over computer networks.

[0005] 2. Background of the Invention

[0006] The health care industry is comprised of a large number of organizations and facilities that vary widely in technical sophistication and capability from the well organized, well staffed and well funded acute care facilities; to individual Doctor's offices; and, in between, to Intermediate/Long Term Care Facilities such as small rural hospitals, psychiatric institutions, nursing homes and assisted living facilities. The latter facilities shall be referred to as Intermediate Care Facilities throughout the remainder of this document.

[0007] Providers of goods and services (sometimes referred to as Vendors) to the healthcare industry have a similar range of size and sophistication.

[0008] Long-term care facilities in the United States generally fall into one of two categories: (i) skilled nursing facilities (SNFs), for residents frail and ill enough to require continuous nursing attention, but not so acutely ill as to require hospitalization; and (ii) assisted living facilities (ALFs), for residents unable to cope with the activities of daily living on their own, but who do not require continuous nursing attention.

[0009] In 1999, approximately $120 billion was spent on care in SNFs, a figure which has been growing at an annual rate of about 10% for at least the last decade. The industry briefly consolidated during the mid-1990s, with the top ten SNF chains (including Beverly, Vencor, ManorCare, Integrated, Mariner, Sun, Genesis, and Lenox) accounting for about 20% of industry revenues. With the passage of the 1997 Balanced Budget Act, however, the Federal government imposed significantly stricter reimbursement policies for Medicare residents, the source of between 25% and 40% of most chains' revenues (an additional 25-50% of revenue generally coming from the Medicaid program). As a result, many of the larger chains have been unable to meet debt obligations which they incurred to pay for acquiring additional facilities and ancillary businesses, and a significant number of bankruptcies have occurred, with five of the largest seven chains (and 1,651 of approximately 17,200 homes nationwide) now in receivership. Even so, with occupancy rates exceeding 90% at most homes, industry experts have described this situation as a mere restructuring, rather than a permanent industry setback.

[0010] In California, state figures report that about $62 per bed per day was spent on products and services supplied by various third-party suppliers and service vendors, including rehabilitation and administrative overhead, capital costs and financing costs. The remainder was spent on labor and direct resident care. A brief listing of the basic products and services required to run even a mid-sized SNF will demonstrate the complexity involved in managing an intermediate healthcare facility. These products and services include, at a minimum:

[0011] Dietary supplies (meat, dairy, grocery and produce)

[0012] Housekeeping supplies

[0013] Laundry and linen supplies

[0014] Enteral supplies and supplements (e.g., G, J and NG tubes, Ensure, Sustacal)

[0015] Ostomy supplies

[0016] Oral pharmacy (averaging 5 prescriptions per resident per month, with many residents taking as many as 12 medications per day)

[0017] IV therapy (including total parenteral nutrition [TPN], antibiotic therapy, pain management and hydration)

[0018] Wound care (including special mattresses, ointments and wound dressings to prevent or treat bedsores, as well as wound care consulting services)

[0019] Durable medical equipment (including wheelchairs, walkers, canes and prostheses)

[0020] Medical supplies

[0021] Radiology (portable x-ray) services

[0022] Laboratory services

[0023] Rehabilitation services (including physical therapy, occupational therapy, and speech pathology)

[0024] Respiratory therapy (including oxygen and related equipment and supplies)

[0025] Power and utilities (electricity, gas, cable, telephone and IT services)

[0026] Grounds keeping, repair and other property, operations and maintenance services

[0027] Social services

[0028] Resident activities

[0029] Because the amount spent on many of these products and services varies significantly on a weekly basis (such as rehabilitation services, emergency medical supplies, radiology and laboratory exams, and food & medications, among others), while the amount spent on other items can remain unchanged for months at a time (e.g., utilities, routine resident supplies, resident activities, housekeeping), a "usual" number of transactions per resident per week per facility can be difficult to establish. However, a reasonable estimate is approximately 10-15 transactions per resident per week, about 6-7 of which occur in the eight key areas representing nearly 80% of the total dollar volume involved: four service areas—pharmacy, laboratory, radiology, and rehabilitation; and four product areas—dietary and housekeeping supplies; durable medical equipment, respiratory and medical supplies; specialty beds and wound care supplies; and enteral and ostomy supplies.

[0030] While spending on information technology has lagged throughout the healthcare sector, rarely exceeding 5% of revenues (vs. more than 12% in financial services, for example), the long-term care segment has historically been an area which particularly under-invested in information technology. Many facilities and chains had no computers at all until the Health Care Financing Administration (HCFA) began requiring on-line transmission of the Minimum Data Set (MDS) information, discussed more hereinafter, during the late '90s. The computers generally acquired at that time are simply unable to manage the level of information complexity and on-line access required by today's integrated software packages. At the same time, the operating environment for skilled nursing facilities and other long-term care institutions has rapidly become one of the most complex faced by any industry segment across the entire economy.

[0031] There are currently 4,956 community-based acute-care medical/surgical hospitals in the United States, including approximately 600 psychiatric hospitals. Total spending at these hospitals was nearly $320 billion in 1998, the latest year for which figures are available. The average daily census was 526,000, with approximately 820,000 beds available.

[0032] Compared to SNFs, small hospitals tend to spend more money on labor and administration, and somewhat less on procurement—although absolute spending on goods and services by small hospitals is more than double that accounted for by SNFs.

[0033] Because of the Balanced Budget Act of 1997 (BBA), large acute-care hospitals have cut spending dramatically, as they seek to offset nearly $36 billion of reduced Federal Medicare and Medicaid reimbursements. Small hospitals, however, have been granted a safe harbor by the Balanced Budget Refinement Act of 1999, which granted an additional $1.3 billion specifically to these facilities, with much of the funding earmarked for improvements in information technology. In addition, small and rural hospitals with 100 or fewer beds have been held harmless with respect to their pre-BBA funding levels until at least 2004.

[0034] About $20 billion was spent on (Assisted Living Facilities (ALF) care in 1999, with the top 30 companies involved in the sector (including Marriott, Sunrise, Alterra, Atria, Emeritus, Holiday, Assisted Living Concepts and American Retirement Corp.) accounting for only about 4% of revenues in a remarkably fragmented industry. Over 95% of ALF residents are private pay.

[0035] Many states do not require ALFs to obtain state certification in order to operate (although this is beginning to change and many states are introducing certification requirements and more are expected to join the fold). Therefore, ALFs generally do not have to meet the same level of operating standards as SNFs and thus, are considerably easier to run (with only 10-15 major suppliers and service vendors per facility, compared with 20-25 for a SNF).

[0036] The California Association of Healthcare Facilities (CAHF) 2000 guidebook lists over 125 separate categories of product and service vendors to skilled nursing facilities, ranging from Accounting to X-Ray. While a precise count of vendors nationwide is impossible, due to overlap with other businesses which also purchase food service or housekeeping, a conservative estimate would place the number of vendors to the intermediate healthcare industry at more than 75,000, or nearly 8 separate vendors per facility. Since there is substantial overlap between vendors and facilities, each facility actually deals with 15-20 vendors on a regular basis, and substantially more on occasion (e.g., for new employee background checks).

SUMMARY OF THE INVENTION

[0037] The invention is directed to methods of providing services to and upgrading information technology capabilities at Intermediate Care facilities.

[0038] The invention helps manage the spectrum of intermediate care and long-term care procurement transactions, and tightly integrate these transactions into an overall management, financial, accounting and billing system. Even more importantly, the invention provides compliance feedback, both procurement and clinical, helping the system run smoothly and more efficiently and helping deliver higher quality and more profitable patient and resident care.

[0039] While many of these vendors are large, national firms (such as SYSCO, for food), others are regional or even only municipal in scope. Directly enrolling such a dispersed customer base would be prohibitively expensive, so, in accordance with the invention, vendors are enrolled in conjunction with the enrollment of their customers. Furthermore, since using the CentraLink system is both affordable and efficient for vendors, larger service and product suppliers will act to enroll their facility customers in CentraLink's network, creating a positive cycle of enrollment and helping to drive market penetration.

[0040] The foregoing and other features, aspects and advantages of the present invention will become more apparent from the following detailed description of the present invention when taken in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0041] The objects, features and advantages of the system of the present invention will be apparent from the following description in which:

[0042] **FIG. 1** is a drawing of facility needs, vendor needs and the parameters of a solution in accordance with one aspect of the invention.

[0043] **FIG. 2** is a block diagram of a system architecture for carrying out one aspect of the invention.

[0044] **FIG. 3** is a block diagram of a network arrangement suitable for implementing the invention at an Intermediate Care Facility or at a vendor facility.

[0045] **FIG. 4** is a block diagram illustrating the hardware and software architecture of a workstation such as might be used in implementing the arrangement of **FIG. 3**.

[0046] **FIG. 5** is an exemplary hardware architecture for implementing a Central Server such as shown in **FIG. 2**.

[0047] **FIG. 6** is a block diagram of an exemplary application software architecture of a server implementation in accordance with another aspect of the invention.

[0048] **FIG. 7** is a block diagram of an exemplary software implementation of an Acute Care Subsystem of application programs as shown in **FIG. 6**.

[0049] **FIG. 8** is a block diagram of an exemplary software implementation of a Financial and Accounting Subsystem of application programs as shown in **FIG. 6**.

[0050] **FIG. 9** is a block diagram of an exemplary software implementation of an Intermediate Care Subsystem of application programs as shown in **FIG. 6**.

[0051] **FIG. 10** is a high level flow chart of an exemplary process for ordering supplies and services.

[0052] **FIG. 11** is a high level flow chart of an exemplary process for shipping supplies and delivering services.

[0053] **FIG. 12** is a high level flow chart of an exemplary process for converting MDS data into a searchable database for identifying potential clinical trial candidates and for determining product utilization.

[0054] **FIG. 13** is an illustration of a rules hierarchy for illustrating rules inheritance in accordance with one aspect of the invention.

[0055] **FIGS. 14A and 14B** illustrate high level information flow before and after implementation of the invention, respectively.

[0056] **FIG. 15** is a representation of exemplary benefits provided to users in accordance with one aspect of the invention.

[0057] **FIG. 16** is a diagram showing high level information flow using the central server.

[0058] **FIG. 17** is a block diagram showing the relationship among subsystem modules in accordance with one aspect of the invention.

[0059] **FIG. 18** is a block diagram of an Integrated Compliance Program in accordance with one aspect of the invention.

[0060] **FIG. 19** is a block diagram of an exemplary Information Flow through a procurement process in accordance with one aspect of the invention.

[0061] **FIG. 20** is a comparison of selected features of the invention against application service providers of the prior art.

DETAILED DESCRIPTION OF THE INVENTION

[0062] Applicants have recognized that Intermediate Care Facilities share certain common problems that permit a solution to be crafted that can be adapted to the culture of each individual institution while still accommodating the needs of the universe of Intermediate Care Facilities.

[0063] **FIG. 1** is a drawing of facility needs, vendor needs and the parameters of a solution in accordance with one aspect of the invention. As illustrated in **FIG. 1**, Intermediate Care Facilities (ICFs) are typically cash strapped, with obsolete technology. They possess a variety of dated and certainly incompatible legacy systems requiring massive, wasteful redundant data entry. ICFs typically lack computer literate personnel. As described more hereinafter, ICFs are facing increasing regulatory and margin pressures. Many facilities and chains had no computers at all until HCFA began requiring on-line transmission of the MDS in the late '90s, and the computers generally acquired at that time are simply unable to manage the level of information complexity and on-line access required by today's integrated software packages.

[0064] Vendors to the healthcare industry face some similar pressures. As also illustrated in **FIG. 1**, they too, are cash strapped and have obsolete technology. They are protective of existing customer relationships and are sometimes fearful that automated solutions will displace them from the customer relationships they have carefully built. They are aware that many of their goods and services have become commodities that can be provided by others willing to compete on price. At they same time they are anxious to increase their share of the market.

[0065] **FIG. 2** is a block diagram of a system architecture for carrying out one aspect of the invention. Item **200** represents the Central Server which interlinks a plurality of Intermediate Care Facilities **210** and a plurality of vendors **220**. Although the Intermediate Care Facilities and vendors are shown connected to the Central Server and a star architecture, any type of network connection, for example, token ring, can be utilized to connect the Intermediate Care Facilities **210**, vendors **220** with a Central Server **200**. In a preferred embodiment, the connection between the individual Intermediate Care Facilities and vendors to the Central Server occurs over a virtual private network.

[0066] **FIG. 3** is a block diagram of a network arrangement suitable for implementing the invention at an Intermediate Care Facility or at a vendor facility. As shown in **FIG. 3**, a workstation configured as a wireless hub **300** connects to the Central Server **200** over a network. The wireless hub service is a central node for a wireless local area network interconnecting a plurality of workstations **310** with the Central Server over the wireless hub. The wireless hub also interconnects the workstations **310** with one or more printers **320**. The wireless LAN is preferred in most environments where cabling for an existing network is inadequate to support the installation of the invention. Using wireless LANs permits one to avoid the cost of installing a new wiring plant. In installations where the existing network cabling is sufficient to support LAN operation over optical or over conductive-based medium such as coax or copper, the workstations **310** can be linked to the network control workstation **300** using standard networking technology. A

configuration very similar to that shown in **FIG. 3** is utilized at vendor installations except that typically, a vendor installation will require fewer workstations.

[0067] **FIG. 4** is a block diagram illustrating the hardware and software architecture of a workstation such as might be used in implementing the arrangement of **FIG. 3**. In **FIG. 4**, personal computer **400** is a workstation of, for example, the Intel Pentium Class. Such a workstation has an operating system **410** which, in one embodiment is comprised of the Windows 2000 operating system. It also includes a local area network interface **420** and virtual private network software **430** to enable the workstation to link to the Central Server in a secure manner. A browser **440**, which could in a particular implementation be an Internet Explorer type browser, provides the principal interface to the user when connecting to the Central Server. The variety of other applications **450** may be installed to suit the personal needs of the user of personal computer **400**. When the interconnection that an Intermediate Care Facility or a vendor orders is a wireless LAN connection, the LAN interface **420** will be a wireless LAN interface. When it is a standard network interface, it will be a non-wireless LAN interface. When the personal computer **400** is configured to be the main connection point with the Central Server, the computer is additionally optionally equipped with a hidden local replica **460** of the Central Server functionality and database to permit the terminals at the Intermediate Care Facility to function, notwithstanding the link to the Central Server might go down. If, in fact, the Central Server goes down, the individual terminals can continue to operate with the hidden local replica until such time as the link is restored. At that time, the Central Server will synchronize the Central Server database with the transactions and information that has been stored in the hidden local replica and the information at the Central Server will thereafter be updated in real time.

[0068] **FIG. 5** is an exemplary hardware architecture for implementing a Central Server such as shown in **FIG. 2**. Storage area network **500** comprises a plurality of Compaq DV580 servers running Microsoft SQL 2000 server software. They are connected in any one of several feasible configurations to constitute the storage area network. The interface between the storage area network and the main network **515** is through cache server **510**. The cache server **510** stores replicas of pages within the storage area network to facilitate their rapid retrieval if they are used more than once in a well-known fashion.

[0069] A plurality of application servers **520** operate in a load sharing mode and provide services to users over the network **515**. The interface to the network from the external world is fully redundant. The interface server **530** maintains separate firewalls **540** going to separate ISPs over ISP interfaces **550**. The ISPs are connected via separate routers **560** and by separate physical paths maintained by separate carriers. The application servers are typically compact DV360 class dual processor class of devices running Windows 2000 operating system and Microsoft application server software. Services are delivered to end users utilizing Citrix server software on the Central Server side and by using a Citrix client on the individual workstations of the vendors and Intermediate Care Facilities. An R&D/test server environment **570** is maintained to enable new software implementations to be tested without impacting operational functionality. FTP servers **580** permit materials to be received and downloaded from end user workstations utilizing File Transfer Protocol. A network operation center contains overall system management software such as Syslog, Link Tools, Compaq Instant Manager, Net IQ, Wats Up and RMS Console. Any number of network maintenance and observation tools may be utilized to ensure the network is up and running and fully functional at any particular point in time.

[0070] **FIG. 6** is a block diagram of an exemplary application software architecture of a server implementation in accordance with another aspect of the invention. The software architecture for the Central Server hardware described in conjunction with the previous Figure comprises three subsystems. The Acute Care Subsystem **600** is dominated with that nomenclature because it shows in common some functionality required by Acute Care Institutions. However, the invention is directed to the Intermediate Care Facility market and not to the Acute Care Market. The Intermediate Care Subsystem **610** contains assertive software to be described hereinafter as those the financial/accounting subsystem **620**.

[0071] Each of these subsystems will be described more hereinafter and is described in detail in the CD ROM Appendices attached hereto.

[0072] **FIG. 7** is a block diagram of an exemplary software implementation of an Acute Care Subsystem of application programs as shown in **FIG. 6**. The Acute Care Subsystem comprises a patient module **700** which deals mainly with the demographics, admissions, discharge, transfer and current census of patients within the Intermediate Care Facility. The clinical patient management module **710** includes software for allowing a physician to enter orders with respect to a patient, for charting a patient, for creating nursing care plans, for entering and recording standing orders, for providing targets and goals for a patient's care and for providing a treatment care profile. The information in this module is utilized to create a workflow for a nurse assigned to care for a particular patient and to aggregate the information for a particular patient with that of other patients assigned to the care of that nurse so that the nurse has an integrated view of the workflow needed to carry out the proper care of patients within her jurisdiction. This is a rules based system and the data entered by the various modules results in triggering appropriate rules which implement the functionality. The exemplary rules for carrying out the invention are shown in the attached CD ROM Appendices. In addition to generating the workflows for a particular nurse, the rules based system also provides output to the financial and accounting subsystem so that appropriate billing and payment can be accounted for.

[0073] **FIG. 8** is a block diagram of an exemplary software implementation of a Financial and Accounting Subsystem of application programs as shown in **FIG. 6**. The financial/accounting subsystem comprises a plurality of modules such as accounts receivable, accounts payable, billing, general ledger and the like, which are routine and well-known in the healthcare industry.

[0074] **FIG. 9** is a block diagram of an exemplary software implementation of an Intermediate Care Subsystem of application programs as shown in **FIG. 6**. The Intermediate Care Subsystem includes a variety of software modules including electronic procurement, vendor compliance, clini-

cal compliance, clinical trials, and MDS Manager. These modules are described more hereinafter and in the associated CD ROM Appendices attached hereto.

[0075] **FIG. 10** is a high level flow chart of an exemplary process for ordering supplies and services. This flowchart describes a process which can be utilized to order supplies or services. At step **1000**, an optional check of the inventory management subsystem indicates that supplies are low. Alternatively, a rule may fire when an item in inventory reaches a threshold level, alerting a user that an order needs to be placed. At step **1010**, Intermediate Care Facility purchasing personnel logs into the purchasing module and enters a class of goods or services to be ordered. A list of authorized suppliers for the Intermediate Care Facility is displayed together with ordering information at step **1020**. Optionally, step **1030**, the ICF ordering personnel can view the compliance information on a particular vendor and compare the compliance information with other vendors who supply the same goods or services to determine the appropriate destination for the order. Once the order is completed, step **1040**, the ordering information is submitted by the selected vendor and the compliance information updated. In step **1050**, that information on the status of the order entered in the database and/or the financial subsystem to prevent appropriate billing and payment records to be generated.

[0076] **FIG. 11** is a high level flow chart of an exemplary process for shipping supplies and delivering services. Before shipping an order or providing services, the vendor may optionally view the account information status of the Intermediate Care Facility (step **1100**). If appropriate, the vendor ships the order or delivers the service (**1110**). The vendor then enters completion information in the database and/or the financial subsystem (**1120**) and enters appropriate billing information for the ICF (**1130**).

[0077] **FIG. 12** is a high level flow chart of an exemplary process for converting MDS data into a searchable database for identifying potential clinical trial candidates and for determining product utilization. At the ICF, a copy of the MDS data from the facility is made (**1200**) and cleansed or sanitized to remove data from the MDS records or hit the guidelines (**1210**). The cleansed MDS file and transferred from the facility to the Central Server over a network (**1220**) and when the MDS file is received at the Central Server (**1230**), the individual's records are read and inserted into a database where database records are updated and records are marked for analysis.

[0078] The more updated records are transferred to a query database table which is utilized as the object for information retrieval queries by users (**1240**). A user can then query the query database table for potential clinical trial candidates and/or for product utilization (**1250**).

[0079] **FIG. 13** is an illustration of a rules hierarchy for illustrating rules inheritance in accordance with one aspect of the invention. The rules utilized to implement the invention eacg have a scope of application. Rules at a lower level in the hierarchy may inherit characteristics of rules higher in the hierarchical level. For example, as shown in **FIG. 13**, a plurality of rules may have system-wide application. These rules may be inherited by a variety of enterprises and sub-enterprises. For example, North America may constitute an enterprise having two sub-enterprises of Canada and the United States. Canada, having a socialized healthcare system, divides the enterprises by province so that each province, as a sub-sub enterprise, may have its own rules.

[0080] In the United States of America, on the other hand, the rules may be unique to a particular healthcare enterprise, such as Global Health or Columbia Health, illustrated in **FIG. 13**. Columbia Health, for example, may have East Coast and West Coast sub-sub-enterprises and the West Coast sub-sub-sub-enterprise may have a sub-sub-sub-sub-enterprise for California having a plurality of facilities such as hospital **1** and long-term care facility **16** and psychiatric hospital **2**. The facilities may each have a plurality of institutions within the facilities, such as a long-term care unit, clinic **3** and clinic **4** for psychiatric hospital **2**. In short, local rules at any level of the hierarchy may be instantiated by inheritance from rules above or may be customized for the institution, facility or enterprise level with which they are associated.

[0081] **FIGS. 14A and 14B** illustrate at a high level procurement information flow before and after implementation of the invention.

[0082] According to the Gartner Group, electronic business-to-business procurement is likely to increase from $145 billion in 1999 to over $7.3 trillion in 2004. While other researchers offer somewhat lower numbers (such as $3.0 trillion in 2004, according to the Yankee Group), the e-procurement opportunity is undoubtedly large across industries. Simply by reducing the rogue purchasing associated with antiquated catalog and paper-based procurement, many companies (including intermediate healthcare facilities) have discovered that they can immediately decrease costs between 5 and 15%. For some facilities and chains, the number has proved to be as high as 20-40%. Furthermore, many of the personnel ordering resident and institutional goods in the intermediate healthcare setting now do so with inadequate training, with inadequate or contradictory resident information, and with significant under-staffing. By hard-wiring sensible procurement choices into the options presented to these personnel, the invention's convenient, reliable and comprehensive ordering system enforces pre-established formularies and contracting criteria, and creates substantial value both for facilities and vendors.

[0083] **FIG. 15** is a representation of exemplary benefits provided to users in accordance with one aspect of the invention. As shown in **FIG. 15**, data from facility operations is sent (**1500**) to the Central Server. The Central Server provides software support (**1510**) to the facility operations. The Central Server processes the operational data from the facilities and provides a variety of value added feedback to management about the operation of the facility and about compliance by vendors and about clinical compliance, thus optimizing the income from the facility and optimizing compliance with external regulations to minimize administrative difficulties from regulators.

[0084] **FIG. 16** is a diagram showing high level information flow using the central server. Some benefits to the facility utilizing the Central Server as shown in more detail in **FIG. 16**. Clinical data, hospital data and procurement data are all provided to the Central Server. On the facility side, modules track census, MDS management, clinical compliance, contact compliance, generate care plans, create operational scenarios and provide billing and cash-flow informa-

tion to a financial/accounting module and back to the central control where the inventory status is monitored permitting the procurement cycle to be initiated appropriately when supplies or services are low.

[0085] FIG. 17 is a block diagram showing the relationship among subsystem modules and revenue streams in accordance with one aspect of the invention.

[0086] Three key factors enable the invention to offer this comprehensive solution to its customers and users. First, the intermediate healthcare market is significantly less complex than the acute care market, where integrators have repeatedly tried and failed to master the overwhelming complexity of the sector. By contrast, intermediate healthcare facilities rely on well-established algorithms to monitor operations compliance, contract compliance, and the associated accounting and billing tasks. Thus, the integration task within this market is more like that in the traditional small and medium-sized enterprise (SME) market, where integration has been routinely successful, than the acute care market, where integration has for the most part failed.

[0087] Second, the purchasing decision in intermediate healthcare is significantly less complex than in the acute healthcare environment. Particularly in stand-alone facilities and small chains, the facility owner or empowered administrator is responsible for nearly all procurement, including management information systems.

[0088] Finally, the invention focuses on maintaining the linkages and integration between modules, rather than on developing the modules themselves.

[0089] Connectivity fees are one source of revenue. The invention provides a fully-operational hardware, software and networking package to its facility and vendor customers for one low monthly fee, with no up-front investment costs. Market research has shown that facility customers will generally need an average of five (5) workstations, supplemented by one hard-copy printer, a wireless hub, a local router and a high speed (e.g. DSL) connection. Vendors will generally require a similar arrangement, but with only two (2) workstations. Using hardware, software and networking and support services supplied by strategic partners, the entire package can be offered to facilities for a nominal monthly cost.

[0090] An addition revenue stream comes from facility and vendor subscriptions for access to core ASP productivity applications. These applications can be offered according to a cafeteria plan, with several levels of service and associated price. The most basic level of service one can offer to facilities comprises an electronic procurement and contract compliance monitoring applications. The next level of service includes operations compliance monitoring, MDS manager and census manager applications can be added, for an additional amount per month. Finally, facilities may opt for physician order and clinical assessment applications, available for an incremental amount per month. The full ASP management package is available to skilled nursing facilities for an amount well within most administrative budgets.

[0091] FIG. 18 is a block diagram of an Integrated Compliance Program in accordance with one aspect of the invention. Given the demanding nature of healthcare, both contract and operations compliance have long been significant problems for facility administrators. Vendor pricing can routinely vary by up to 80% per SKU (stock-keeping unit), depending on the terms and conditions of a given contract, and persistent confusion among high-turnover staff members usually guarantees that contract compliance as to the terms and conditions of service is often somewhat of a mystery. Because of the wide variability of SKU pricing, the vendor or "contract" compliance monitoring system does not offer full pricing transparency to facility staff, unless requested, but merely checks that the pricing of items within a given contract adheres to that contract's guidelines and specifications. The compliance system also monitors the terms and conditions of services promised against services actually delivered, so that facilities and vendors are better able to understand and measure value at the time contracts are re-negotiated—for example, in determining the window of time during which a "stat" order has actually been delivered. This level of transparency is carefully crafted to benefit both facilities and vendors, since facilities will now have access to an on-going record of actual vendor performance, while high-quality vendors will now be able to rely on an independent record which demonstrates their high quality contract compliance performance.

[0092] An even more important recent driver of change in the healthcare industry has been the need for demonstrated clinical (operations) compliance, both in processes and outcomes. Health department surveyors in most states are authorized to impose a $10,000 fine per instance for any regulatory violation by a skilled nursing facility, and are even authorized to place a facility in immediate receivership, if the situation warrants. Particularly in California, where health department surveyors issue about twice the national average of skilled nursing facility deficiencies, quality of care and regulatory compliance are of the highest priority for every facility administrator. In the same way, JCAHO violations can be critically expensive for small and rural hospitals and psychiatric institutions—precisely where the resources to prepare for surveys are least available and violations most likely to occur.

[0093] In order to gather and maintain information on the physical and mental condition of skilled nursing facility residents, HCFA has created and implemented the Minimum Data Set (MDS), a resident survey instrument that contains 1,800 fields representing 300 demographic and assessment items. In addition to monitoring residents' clinical status, this instrument assists HCFA in determining the specific resource utilization group (RUG) into which a resident will be placed, and accordingly the level of payment that a SNF will receive.

[0094] Both Federal and state governments have begun using HCFA's MDS data to prompt nursing facility surveys, and have significantly increased their funding for surveyors (to over $71 million nationally in 1997 (up 21%), and to more than $7.2 million per year in California alone). But the MDS data alone cannot do more than predict the potential for problems, and intermittent surveys often lead to a "yo-yo" pattern of compliance, with concerns being corrected pending or immediately after a survey and conditions thereafter deteriorating. A recent HCFA study in California found that fewer than three percent of the 493 Los Angeles County skilled nursing facilities that accepted residents covered by Medicaid or Medicare were in full or substantial compliance with all applicable federal standards, with 19 percent having violations that caused actual harm or had the

potential to cause death or serious injury to their residents. And, although California is leading the trend towards more vigorous enforcement, other states have begun to use MDS data to drive inspections as well, signaling a national trend which is top-of-mind for facility administrators nationwide.

[0095] Such widespread difficulties with operations compliance demonstrate that skilled nursing facility administrators are simply being overwhelmed with regulatory, business and staffing pressures. But by integrating their MDS data with a powerful and comprehensive ordering platform, CentraLink offers these facilities an innovative and highly desirable means to support quality care, manage compliance and optimize billing classifications. Specifically, when CentraLink discovers a potential discrepancy in the MDS data submitted from a given skilled nursing facility (for example, an untreated pressure ulcer), it automatically flags the area for consideration of appropriate corrective action (for example, a visit from the facility's wound care supplier).

[0096] Because the invention is both comprehensive and automatic, a facility that contracts for the inventive services and products can address any potential violations well before inspection occurs. In addition, the process flags potential deficiencies as early as possible (since the MDS must be submitted within five days after admission), when they are least expensive to remedy. Finally, by requiring facility approval for the proposed corrective action, the inventive system helps contracting intermediate healthcare administrators tightly focus their budgets while increasing volume for participating suppliers and service vendors—creating a win-win situation between facilities and their key suppliers.

[0097] In addition to vendor and facility compliance, customer satisfaction surveys can be undertaken periodically. As shown in **FIG. 18**, MDS information from a facility is used extensively to manage inventory, to monitor quality and compliance, to check outcome performance, both clinical and financial, and to analyze diagnoses and the resulting cost reimbursement.

[0098] **FIG. 19** is a block diagram of an exemplary Information Flow through a procurement process in accordance with one aspect of the invention. When an order is received (**1900**), the information from the order is placed in the central database (**1905**). The order is routed, typically using email, to the appropriate vendor. The vendor confirms receipt (**1920**) also, preferably by email. When the products or services are delivered by the vendor (**1930**) and the deliver confirmed (**1940**), the transaction is substantially complete. The facility is invoiced by the vendor (**1945**) and an entry made in their accounts payable record. Additionally, an invoice is generated in the accounts receivable column for the vendor. A transaction fee (**1947**) may be charged by the operator the Central Server for the services provided. When billings (**1948**) and collections (**1949**) may be handled by a third party or may be centralized as part of the Central Server activities. When payment is received from a facility or payment made to a vendor (**1950**), the appropriate records are made in the accounting system and money is appropriately transferred. When the delivery is made, compliance and financial information about the order are recorded (**1960**) and can be utilized in reports to the Intermediate Care Facility and to the vendor.

[0099] **FIG. 20** is a comparison of selected features of the invention against application service providers of the prior art. Among healthservice service providers, no other company puts it all together like the inventive system. No other company offers an integrated, networked suite of management applications together with a turn-key connectivity package. No other healthcare ASP integrates e-procurement into their management applications, so no other company can offer the proactive operations compliance monitoring and clinical management features and integrated accounting and billing functions. In addition, since no other company starts out with a networked business model, no other company can offer the integrated and extended applications (e.g., mobile connectivity, improved materials management, etc . . .) that the invention offers. These key market differentiators are illustrated in **FIG. 20**.

[0100] Although the present invention has been described and illustrated in detail, it is clearly understood that the same is by way of illustration and example only and is not to be taken by way of limitation, the spirit and scope of the present invention being limited only by the terms of the appended claims.

Appendix A

CD-ROM Contents

[0101] The following is a description of the contents of 3 CD-ROMs submitted as an appendix to this application:

CD-ROM 1

[0102] This CD-ROM contains 2 Zipped Files. Each zipped file may be opened and the contents viewed using WinZip 7.0 from Nico Mak Computing Inc.

[0103] The first zipped file is 2001-08-25_23-11-36_DB.zip which contains 3 ".dat" files comprising approximately 56 MBytes of uncompressed data. The ".dat" files may be opened and read using Microsoft SQL-2000 software.

[0104] The second zipped file is 2001-08-25_23-11-36_VSS.zip. This zipped file contains 25,521 files comprising approximately 1.3 GBytes of uncompressed data. The extraction index can be viewed using WinZip 7.0. This files contains a variety of file types. The following is a list of file types and a description of how they may be opened/viewed.

File Type	Open/View Using
.doc	Microsoft Word
.logs	Notepad
.a	temp tables used by Microsoft SQL-2000
.b	temp tables used by Microsoft SQL-2000
no extension	
.ini	Notepad
.tmp	not required
.gif	Almost any imaging software

CD-ROM 2

[0105] This CD-ROM contains 7 directories in a Microsoft file directory format. The Directories are:

AP	Subdirectory-Utilities containing 6 files and 4.05 KBytes of data.
	128 files and 1.32 Mbytes of data
AR2	647 files and 15.2 MBytes of data
GL_new	169 files and 2.78 Mbytes of data
Li	357 files and 7.55 Mbytes of data
Se	277 files and 5.38 Mbytes of data
Spirit	77 files and 5.23 Mbytes of data
Ut	197 files and 7.63 Mbytes of data

[0106] The following is a list of file types found in these directories, the subdirectory in which they first appear and a description of how they may be opened/viewed.

File Type	Open/View Using	
In AP:		
"file"	Notepad/Wordpad	These are Providex Programs and need a PVX interpreter to be loaded and read. (same)
.djd	same	
.pvx	same	
.old	same	
.con	same	
.trial	same	
.det	same	
.dhb	same	
In AR2:		
.en	Nomads Library interpreter	
.ldd	same	
.bvh	same	
.slp	same	
.opt	same	
.wch	same	
.ne	same	
.new	same	
In GL_new:		
"bitmap image"	Almost any imaging software	
In Li		
.help	text	
.1st	same	
.cpy	same	
.921	same	
.930	same	
.mai	same	
html	Microsoft HTML Document 5.0	
chart	Microsoft Graph 97	
.asp	same	
.V1	same	
.V2	same	
.001	same	
In Se:		
.dde	same	
.nmd	same	

-continued

File Type	Open/View Using
In Spirit	(all covered previously)
In Ut:	
.utl	same
.pub	same
.bbx	same
.sh	UNIX
.cnv	same
.c	same

CD-ROM 3

[0107] This CD-ROM contains 3 Files comprising approximately 1.7 Mbytes. The following is a list of files-found in these directories and a description of how they may be opened/viewed.

File	Open/View Using
ALF Spec	Microsoft Powerpoint
ASP Design	Microsoft Word
Clinical_Raw_Code	The .xxx file should be renamed .zip and then read as indicated above.

What is claimed is:

1. A method of providing services to an Intermediate Care Facilities (ICF) comprising the steps of:

a. providing a network of workstations at the Intermediate Care Facility;

b. connecting workstations on said network to a remote central server;

c. connecting one or more vendors to said central server; and

d. providing said workstations at the Intermediate Care Facility with access to only ones of said one or more vendors approved by said Intermediate Care Facility.

2. The method of claim 1 in which said network is a wireless network.

3. The method of claim 1 further comprising the step of providing a purchase order from said ICF to one or more of said vendors approved by said ICF.

4. The method of claim 1 further comprising the step of tracking vendor performance on each purchase order.

5. The method of claim 1 further comprising the step of providing reports to said ICF on vendor performance.

6. A method of upgrading information process capability at Intermediate Care Facilities, comprising the steps of:

a. providing workstations at an Intermediate Care Facility;

b. connecting said workstations to a network hub;

c. connecting said network hub to a central applications service provider.

7. The method of claim 6 in which said network hub is a wireless hub.

8. The method of claim 6 in which the applications service provider provides at least one application from the set of

applications consisting of: Vendor Compliance Software, Electronic Procurement Software, Clinical Compliance Software, Clinical Assessment Software, Mobile Connectivity Software, Clinical Trials Software, Census Enhancement Software, MDS Manager Software, Physician Order Software.

9. The method of claim 8 in which the applications service provider further provides at least one application from the set of applications consisting of: Billing Software, Inventory Management Software, Accounts Payable Software, Accounts Receivable Software, Billing Software and Accounting and Financial Software.

10. A method of upgrading information processing capability at Intermediate Care Facilities, comprising the steps of:

a. providing workstations at an Intermediate Care Facility;

b. connecting said workstations to a central applications service provider that provides integrated clinical and procurement applications for the Intermediate Care Facility.

11. The method of claim 10 in which the applications service provider provides at least one application from the set of applications consisting of: Vendor Compliance Software, Electronic Procurement Software, Clinical Compliance Software, Clinical Assessment Software, Mobile Connectivity Software, Clinical Trials Software, Census Enhancement Software, MDS Manager Software, Physician Order Software.

12. The method of claim 10 in which the Intermediate Care Facility may select from a set of clinical and procurement applications and pay a fee based on the number and/or type of applications selected.

13. The method of claim 12 in which the fee is due periodically.

14. A server for connection to at least one Intermediate Care Facility, comprising:

a. a storage area network;

b. a plurality of application servers operating in a load sharing mode;

c. a network interface; and

d. a rules engine for processing information provided by one or more applications.

15. The server of claim 14 in which said rules engine drives one or more of said applications.

16. The server of claim 14 in which said rules engine links data and functions of one or more applications.

17. The server of claim 14 in which said rules engine comprises rules of a rules hierarchy in which rules inherit properties from other rules higher in the hierarchy.

18. The server of claim 14 in which said rules engine comprises rules having a scope of application that applies to an enterprise or subdivision of an enterprise, to facilities and to institutions.

19. The server of claim 14 adapted to be connected to one or more suppliers of goods or services.

20. The server of claim 14 in which said applications include one or more applications from the set of applications consisting of: Vendor Compliance Software, Electronic Procurement Software, Clinical Compliance Software, Clinical Assessment Software, Mobile Connectivity Software, Clinical Trials Software, Census Enhancement Software, Resident Scheduling, MDS Manager Software, Physician Order Software.

21. A method of processing information for use with clinical trials, comprising the steps of:

a. removing patient identifying information from a Minimum Data Set file to produce a revised file;

b. storing information from the revised file in a database; and

c. retrieving information from said database in response to a user query.

22. The method of claim 21 in which said user query is directed to identifying organizations having candidates for participation in a clinical trial.

23. The method of claim 21 in which said user query is directed to identifying organizations having candidates using a particular product or service.

24. A workstation for connection to a central server of an application service provider, said workstation comprising: a hidden copy of data and instructions comprising at least a portion of an application provided by said central server whereby said workstation can continue to provide functionality to users when a communication link to said central server is not operational.

* * * * *

US006675205B2

(12) **United States Patent**
Meadway et al.

(10) **Patent No.: US 6,675,205 B2**
(45) **Date of Patent: Jan. 6, 2004**

(54) **PEER-TO-PEER AUTOMATED ANONYMOUS ASYNCHRONOUS FILE SHARING**

(75) Inventors: **Michael D. Meadway**, Snohomish, WA (US); **Gary W. Tripp**, Bainbridge Island, WA (US)

(73) Assignee: **Arcessa, Inc.**, Kirkland, WA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/910,460**

(22) Filed: **Jul. 20, 2001**

(65) **Prior Publication Data**

US 2002/0073137 A1 Jun. 13, 2002

Related U.S. Application Data

(63) Continuation-in-part of application No. 09/419,405, filed on Oct. 14, 1999, now Pat. No. 6,516,337, and a continuation-in-part of application No. 09/575,971, filed on May 23, 2000.

(60) Provisional application No. 60/219,983, filed on Jul. 21, 2000.

(51) **Int. Cl.**[7] **G06F 13/00**

(52) **U.S. Cl.** **709/219**; 709/203; 709/329; 707/10

(58) **Field of Search** 709/201, 202, 709/203, 217, 219, 313, 329; 707/10, 100, 104

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,093,911	A *	3/1992	Parks et al.	707/1
5,983,216	A *	11/1999	Kirsch et al.	707/2
5,987,506	A *	11/1999	Carter et al.	709/213
6,243,676	B1 *	6/2001	Witteman	704/243
6,253,198	B1 *	6/2001	Perkins	707/3
6,308,173	B1 *	10/2001	Glasser et al.	707/9
6,470,332	B1 *	10/2002	Weschler	707/3

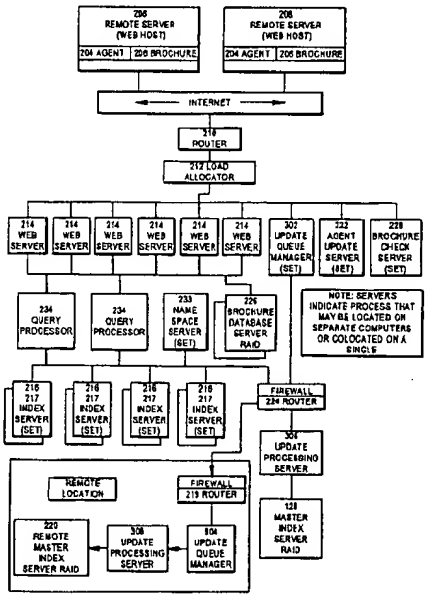
* cited by examiner

Primary Examiner—Viet D. Vu
(74) *Attorney, Agent, or Firm*—Black Lowe & Graham, PLLC

(57) **ABSTRACT**

A service on a computer network which performs centralized searches based on index information transmitted by peer systems to a central site using an agent program running on each peer and then directs the peer systems to each other for the purpose of retrieving files. If none of the peer systems known to contain the file is online (and the file is therefore not available), the request is placed in a queue of file requests maintained by the central site. When a system containing the requested file connects to the service, the requested file is retrieved from that system and then distributed to the other systems which had requested the file. Files retrieved for systems not currently online are held in a queue until the user connects or are emailed to the user, usually as an email attachment. Or, when a computer system containing the file connects to the central site, the file is sent by the system containing the file either to the central site or directly to the user who requested the file via email attachment.

9 Claims, 34 Drawing Sheets



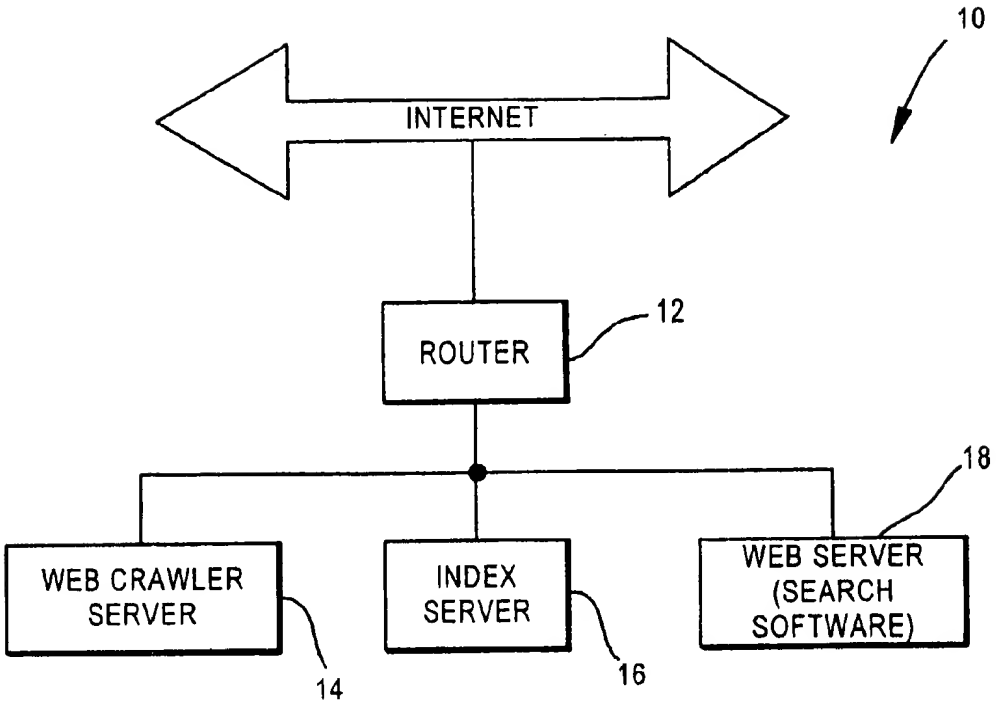


FIG.1
(PRIOR ART)

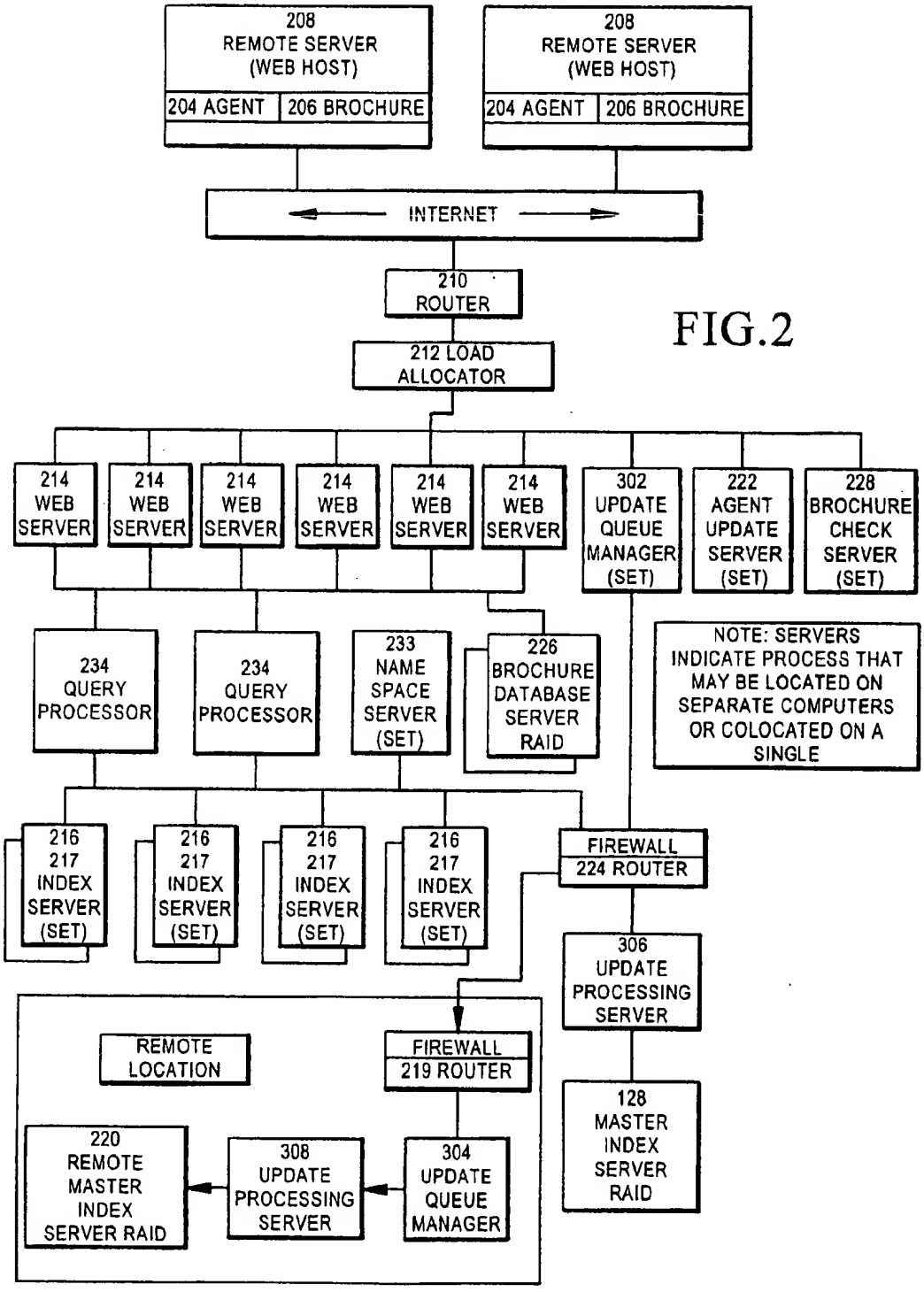

208
REMOTE SERVER
(WEB HOST)
204 AGENT
206 BROCHURE
208
REMOTE SERVER
(WEB HOST)
204 AGENT
206 BROCHURE
INTERNET
210
ROUTER
212 LOAD
ALLOCATOR
214
WEB
SERVER
214
WEB
SERVER
214
WEB
SERVER
214
WEB
SERVER
214
WEB
SERVER
214
WEB
SERVER
302
UPDATE
QUEUE
MANAGER
(SET)
222
AGENT
UPDATE
SERVER
(SET)
228
BROCHURE
CHECK
SERVER
(SET)
234
QUERY
PROCESSOR
234
QUERY
PROCESSOR
233
NAME
SPACE
SERVER
(SET)
226
BROCHURE
DATABASE
SERVER
RAID
NOTE: SERVERS
INDICATE PROCESS THAT
MAY BE LOCATED ON
SEPARATE COMPUTERS
OR COLOCATED ON A
SINGLE
216
217
INDEX
SERVER
(SET)
216
217
INDEX
SERVER
(SET)
216
217
INDEX
SERVER
(SET)
216
217
INDEX
SERVER
(SET)
FIREWALL
224 ROUTER
306
UPDATE
PROCESSING
SERVER
REMOTE
LOCATION
FIREWALL
219 ROUTER
128
MASTER
INDEX
SERVER
RAID
220
REMOTE
MASTER
INDEX
SERVER RAID
308
UPDATE
PROCESSING
SERVER
304
UPDATE
QUEUE
MANAGER

FIG.2

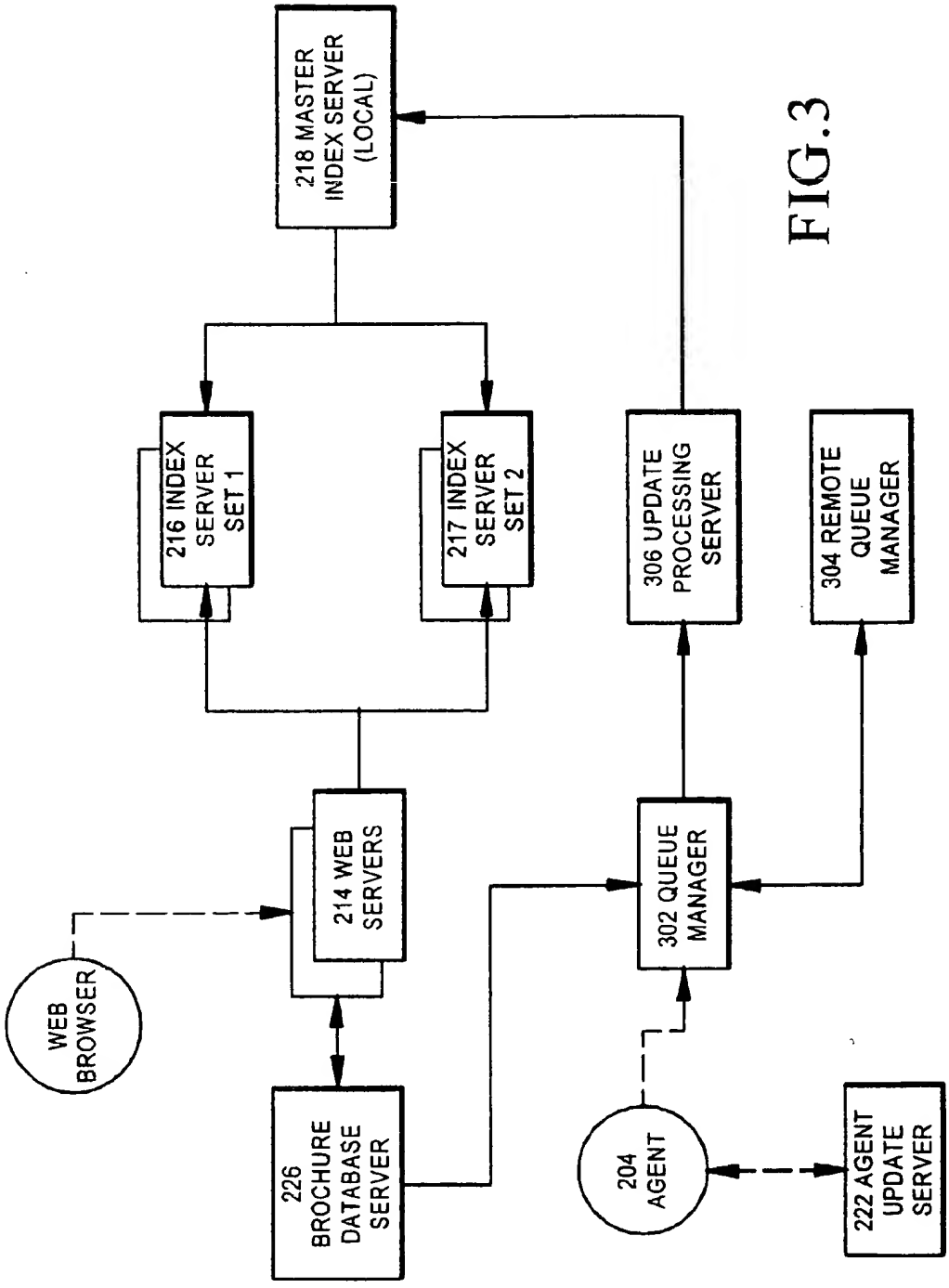

218 MASTER INDEX SERVER (LOCAL)
216 INDEX SERVER SET 1
217 INDEX SERVER SET 2
306 UPDATE PROCESSING SERVER
304 REMOTE QUEUE MANAGER
214 WEB SERVERS
302 QUEUE MANAGER
WEB BROWSER
226 BROCHURE DATABASE SERVER
204 AGENT
222 AGENT UPDATE SERVER

FIG.3

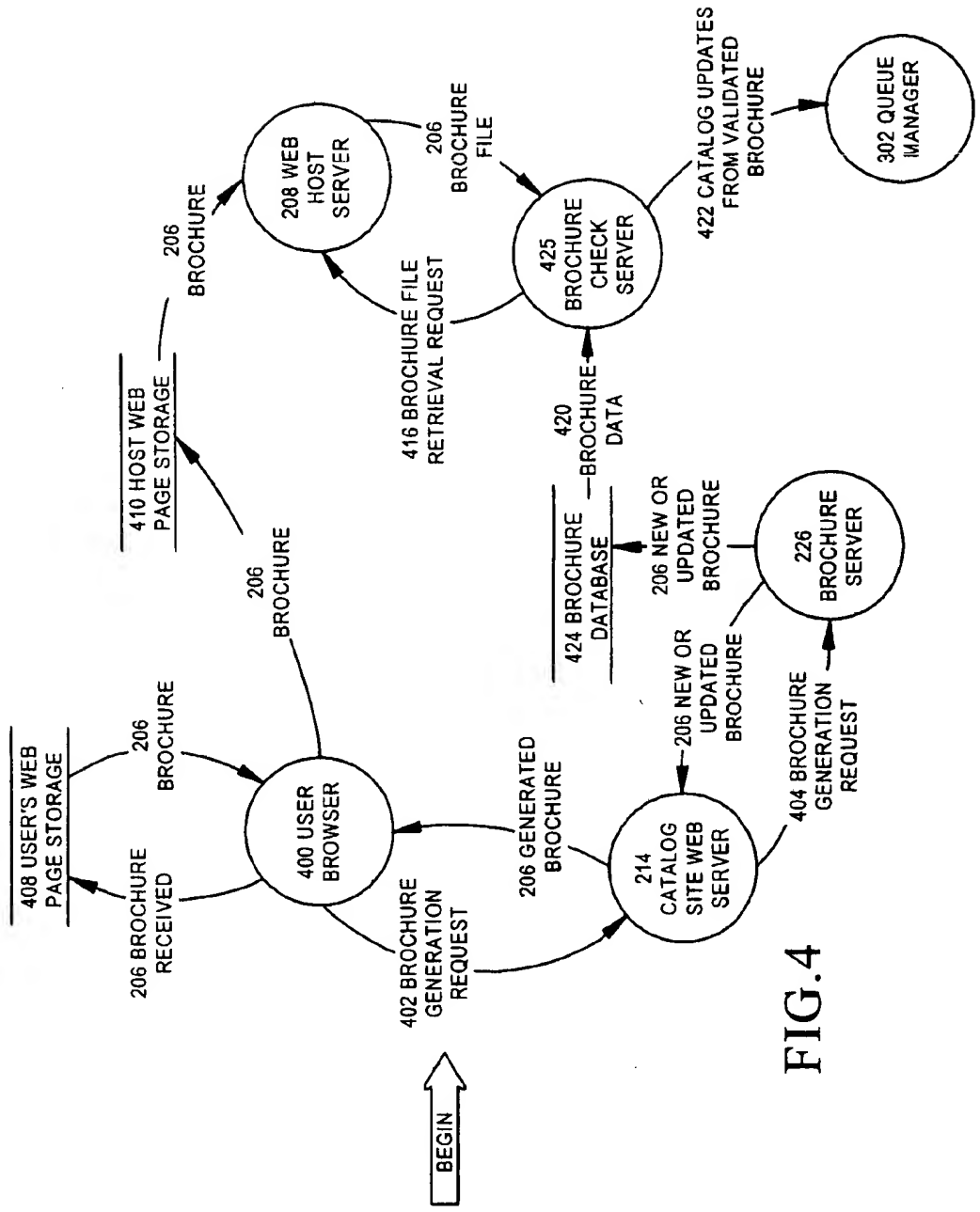
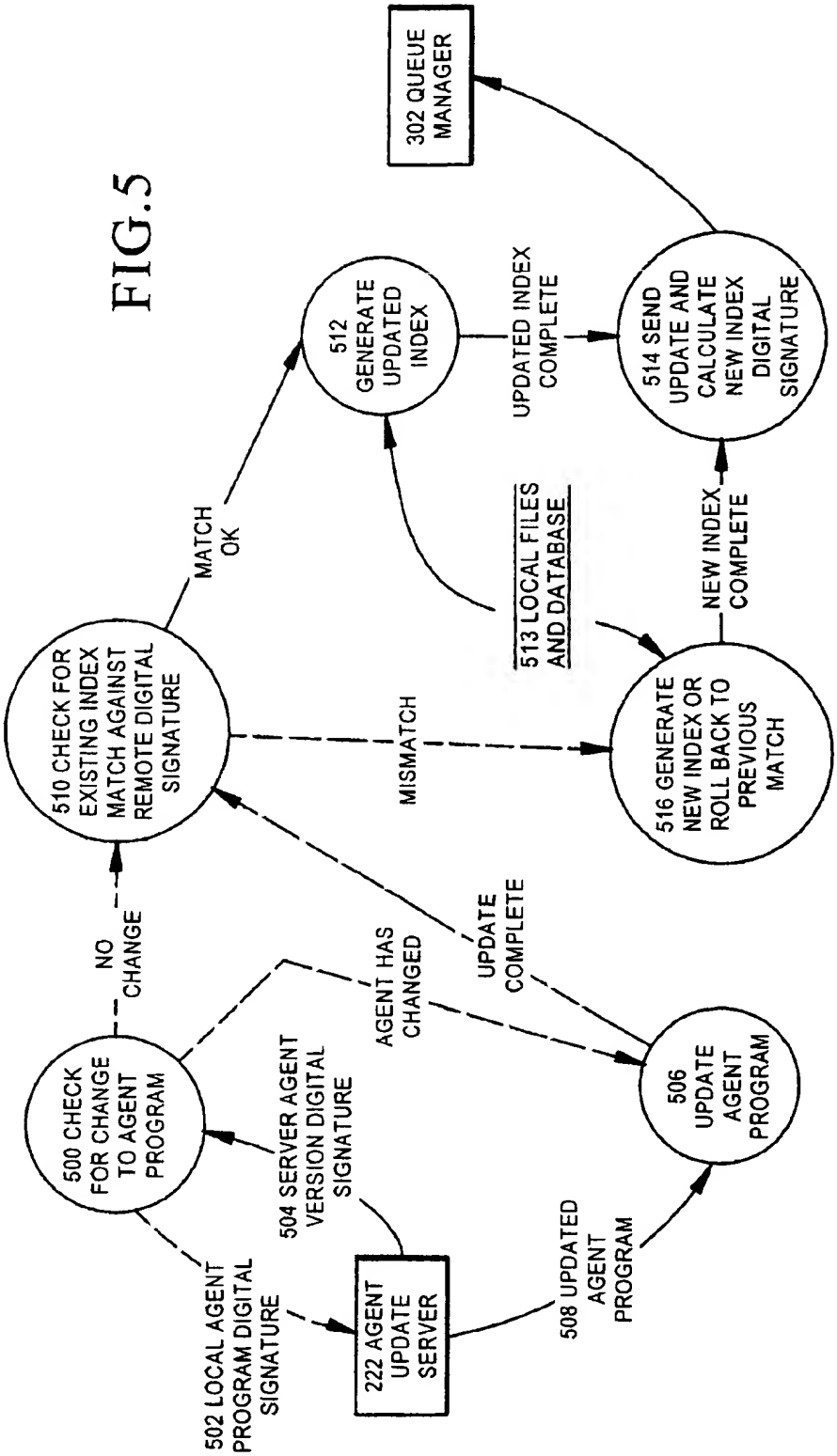

BEGIN
400 USER BROWSER
214 CATALOG SITE WEB SERVER
208 WEB HOST SERVER
425 BROCHURE CHECK SERVER
226 BROCHURE SERVER
302 QUEUE MANAGER
408 USER'S WEB PAGE STORAGE
410 HOST WEB PAGE STORAGE
424 BROCHURE DATABASE
206 BROCHURE
206 BROCHURE RECEIVED
206 BROCHURE
206 BROCHURE
206 BROCHURE FILE
206 GENERATED BROCHURE
206 NEW OR UPDATED BROCHURE
206 NEW OR UPDATED BROCHURE
420 BROCHURE DATA
422 CATALOG UPDATES FROM VALIDATED BROCHURE
402 BROCHURE GENERATION REQUEST
404 BROCHURE GENERATION REQUEST
416 BROCHURE FILE RETRIEVAL REQUEST

FIG.4

FIG.5


302 QUEUE MANAGER
512 GENERATE UPDATED INDEX
UPDATED INDEX COMPLETE
514 SEND UPDATE AND CALCULATE NEW INDEX DIGITAL SIGNATURE
MATCH OK
513 LOCAL FILES AND DATABASE
NEW INDEX COMPLETE
510 CHECK FOR EXISTING INDEX MATCH AGAINST REMOTE DIGITAL SIGNATURE
MISMATCH
516 GENERATE NEW INDEX OR ROLL BACK TO PREVIOUS MATCH
NO CHANGE
AGENT HAS CHANGED
UPDATE COMPLETE
500 CHECK FOR CHANGE TO AGENT PROGRAM
504 SERVER AGENT VERSION DIGITAL SIGNATURE
506 UPDATE AGENT PROGRAM
502 LOCAL AGENT PROGRAM DIGITAL SIGNATURE
222 AGENT UPDATE SERVER
508 UPDATED AGENT PROGRAM

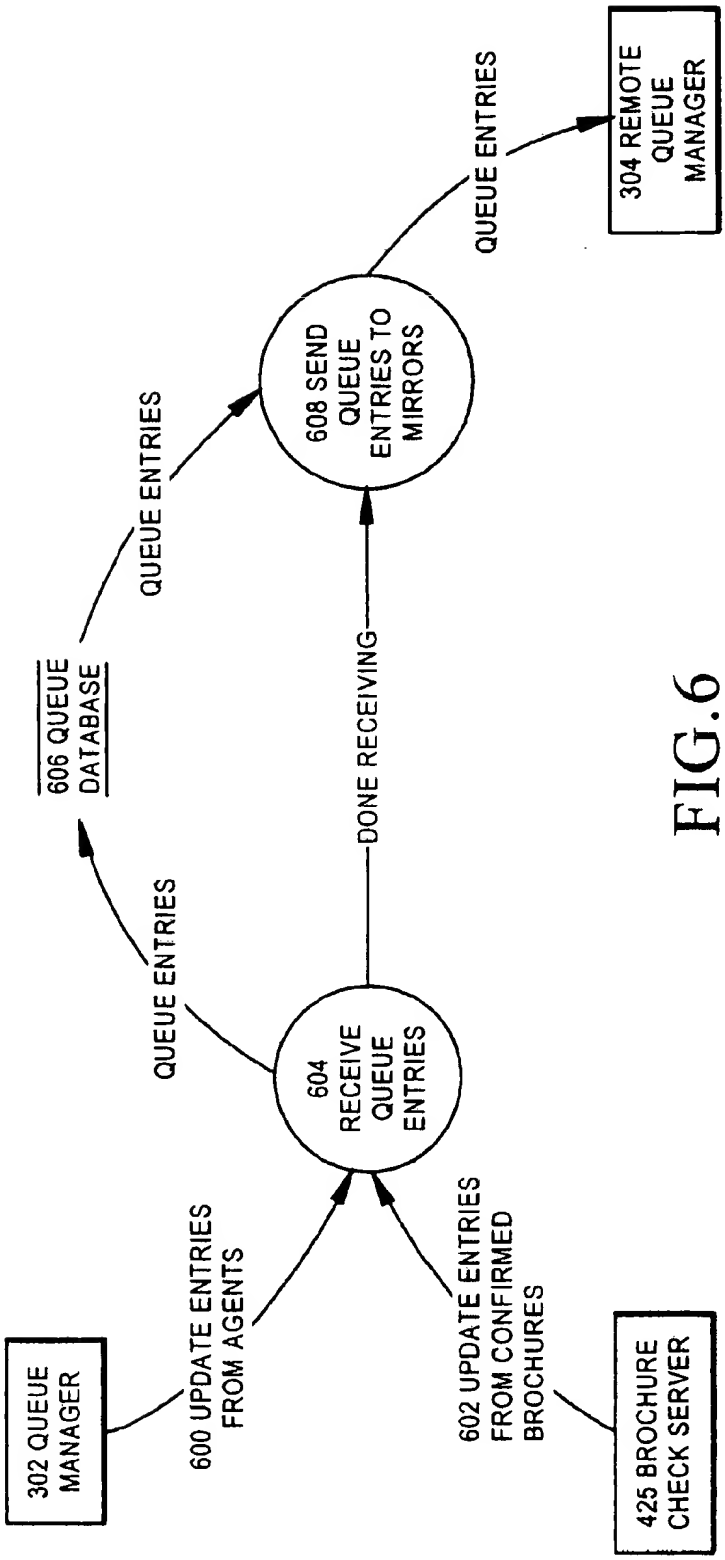

302 QUEUE MANAGER
600 UPDATE ENTRIES FROM AGENTS
602 UPDATE ENTRIES FROM CONFIRMED BROCHURES
425 BROCHURE CHECK SERVER
604 RECEIVE QUEUE ENTRIES
QUEUE ENTRIES
606 QUEUE DATABASE
DONE RECEIVING
608 SEND QUEUE ENTRIES TO MIRRORS
QUEUE ENTRIES
304 REMOTE QUEUE MANAGER

FIG.6

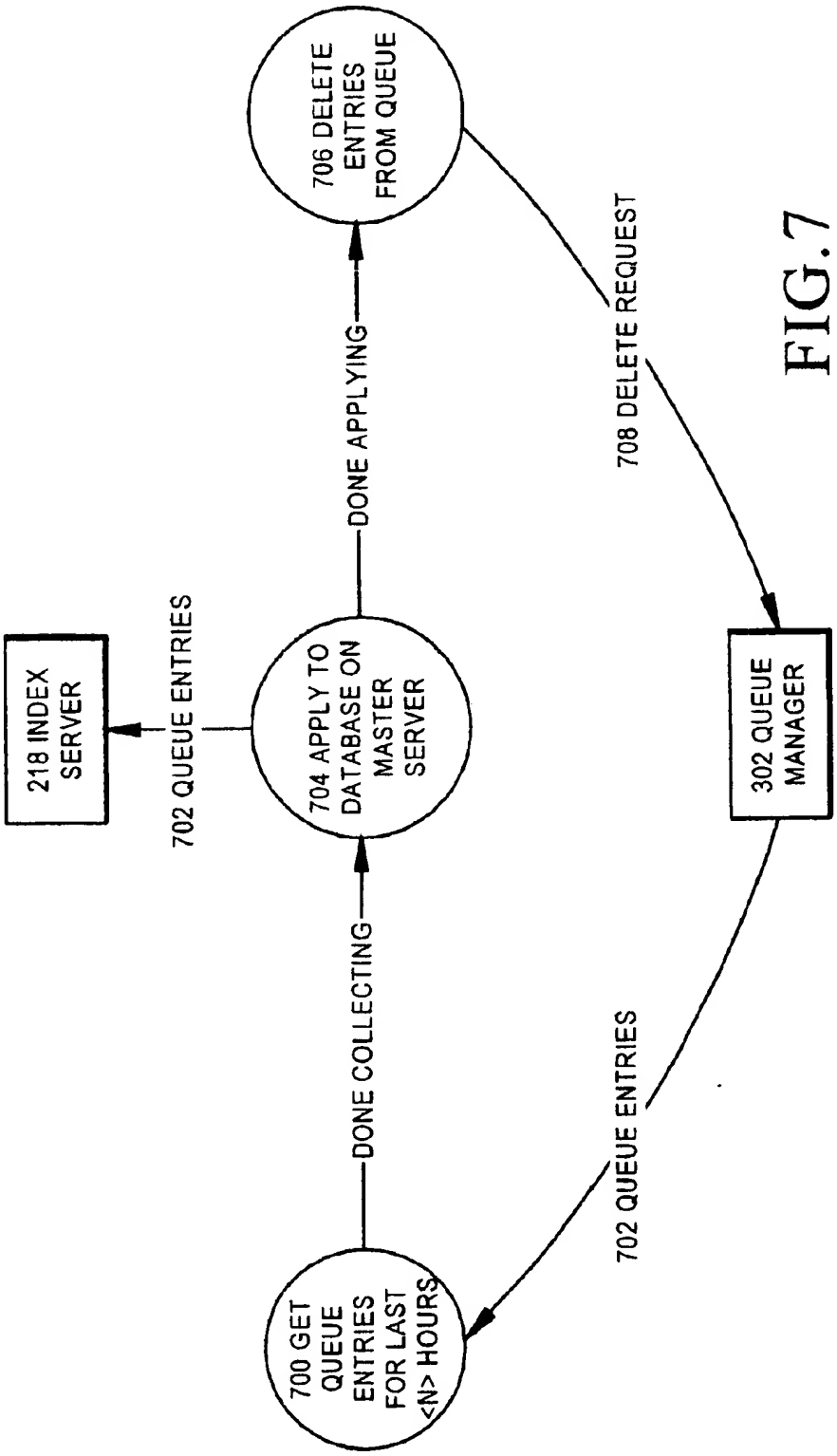
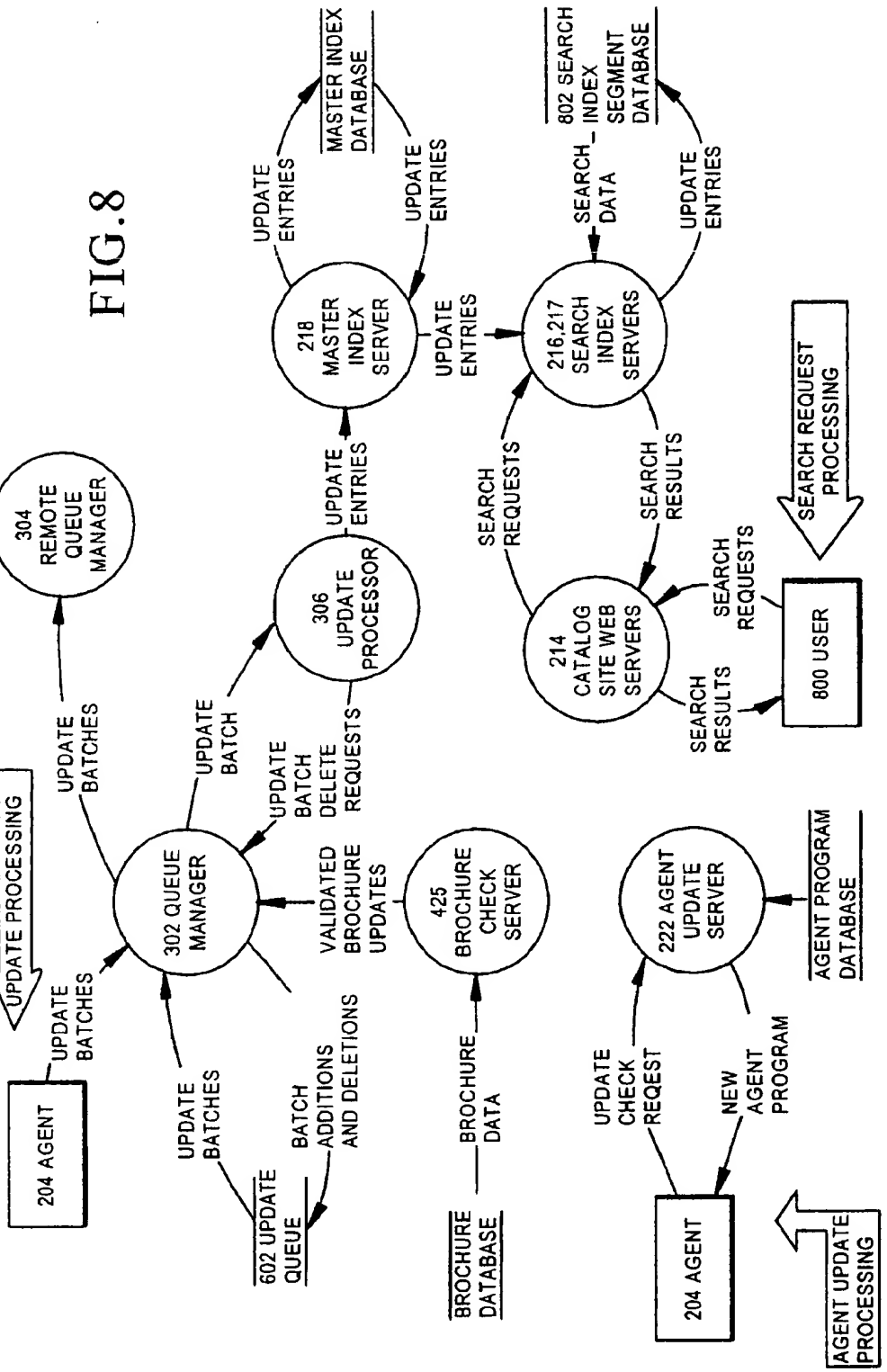

706 DELETE ENTRIES FROM QUEUE
708 DELETE REQUEST
DONE APPLYING
218 INDEX SERVER
702 QUEUE ENTRIES
704 APPLY TO DATABASE ON MASTER SERVER
302 QUEUE MANAGER
DONE COLLECTING
702 QUEUE ENTRIES
700 GET QUEUE ENTRIES FOR LAST <N> HOURS

FIG. 7

FIG. 8


UPDATE PROCESSING
204 AGENT
UPDATE BATCHES
UPDATE BATCHES
304 REMOTE QUEUE MANAGER
302 QUEUE MANAGER
UPDATE BATCHES
602 UPDATE QUEUE
BATCH ADDITIONS AND DELETIONS
UPDATE BATCH
UPDATE BATCH DELETE REQUESTS
306 UPDATE PROCESSOR
VALIDATED BROCHURE UPDATES
425 BROCHURE CHECK SERVER
BROCHURE DATA
BROCHURE DATABASE
UPDATE ENTRIES
MASTER INDEX DATABASE
UPDATE ENTRIES
218 MASTER INDEX SERVER
UPDATE ENTRIES
UPDATE ENTRIES
802 SEARCH INDEX SEGMENT DATABASE
SEARCH DATA
216,217 SEARCH INDEX SERVERS
SEARCH REQUESTS
SEARCH RESULTS
SEARCH REQUEST PROCESSING
214 CATALOG SITE WEB SERVERS
SEARCH REQUESTS
SEARCH RESULTS
800 USER
UPDATE CHECK REQEST
NEW AGENT PROGRAM
222 AGENT UPDATE SERVER
AGENT PROGRAM DATABASE
204 AGENT
AGENT UPDATE PROCESSING

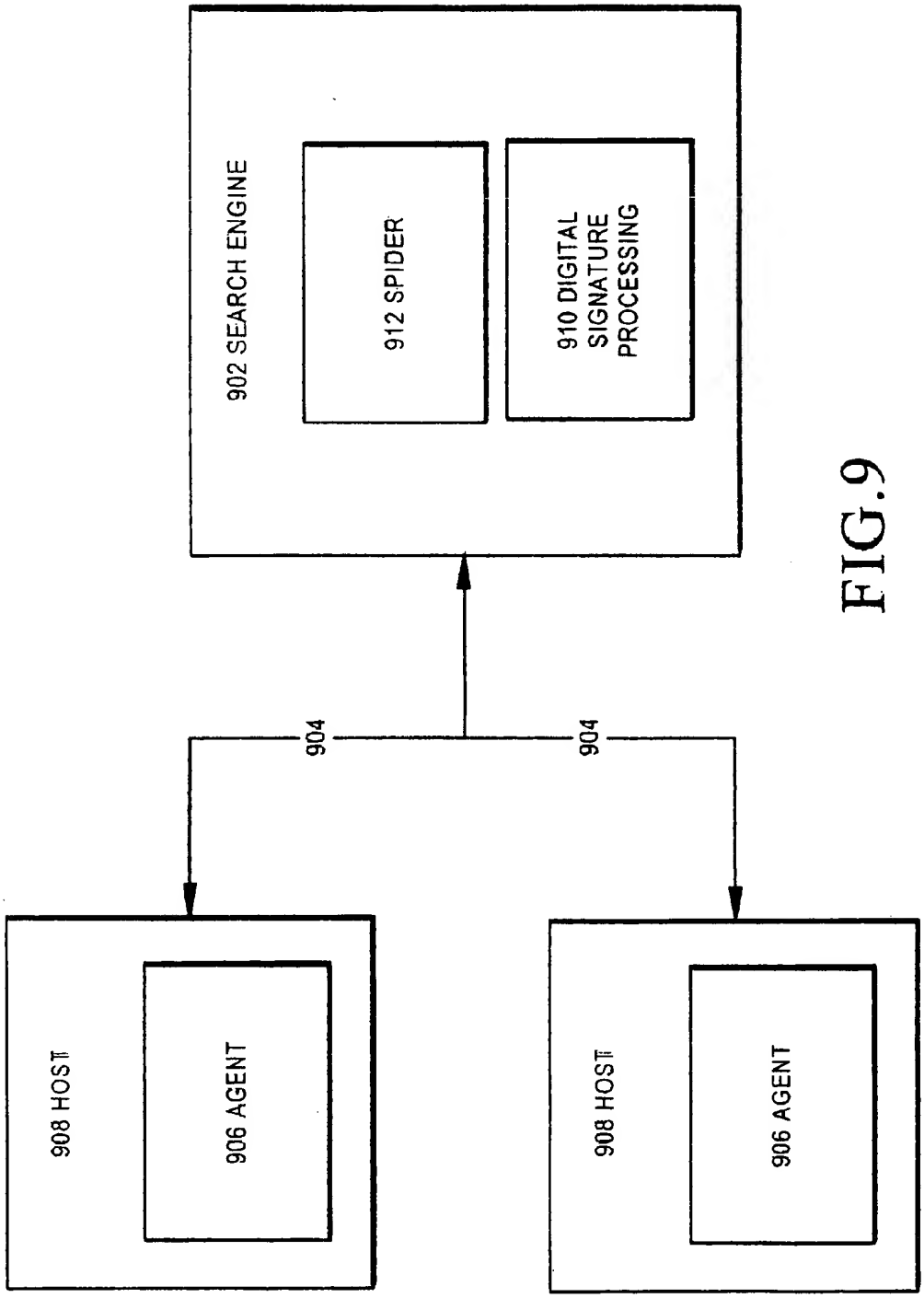

902 SEARCH ENGINE
912 SPIDER
910 DIGITAL SIGNATURE PROCESSING
904
904
908 HOST
906 AGENT
908 HOST
906 AGENT

FIG.9

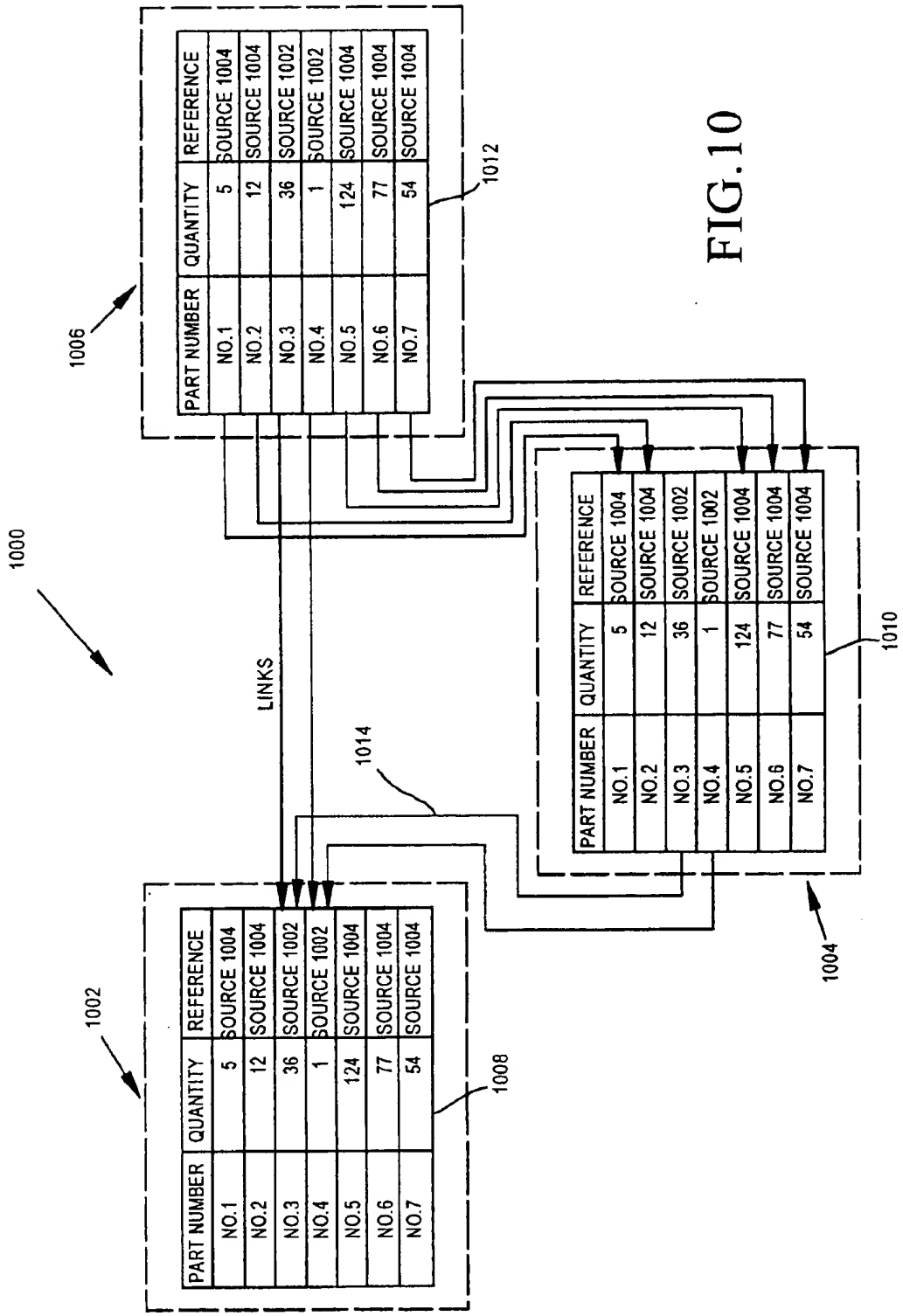
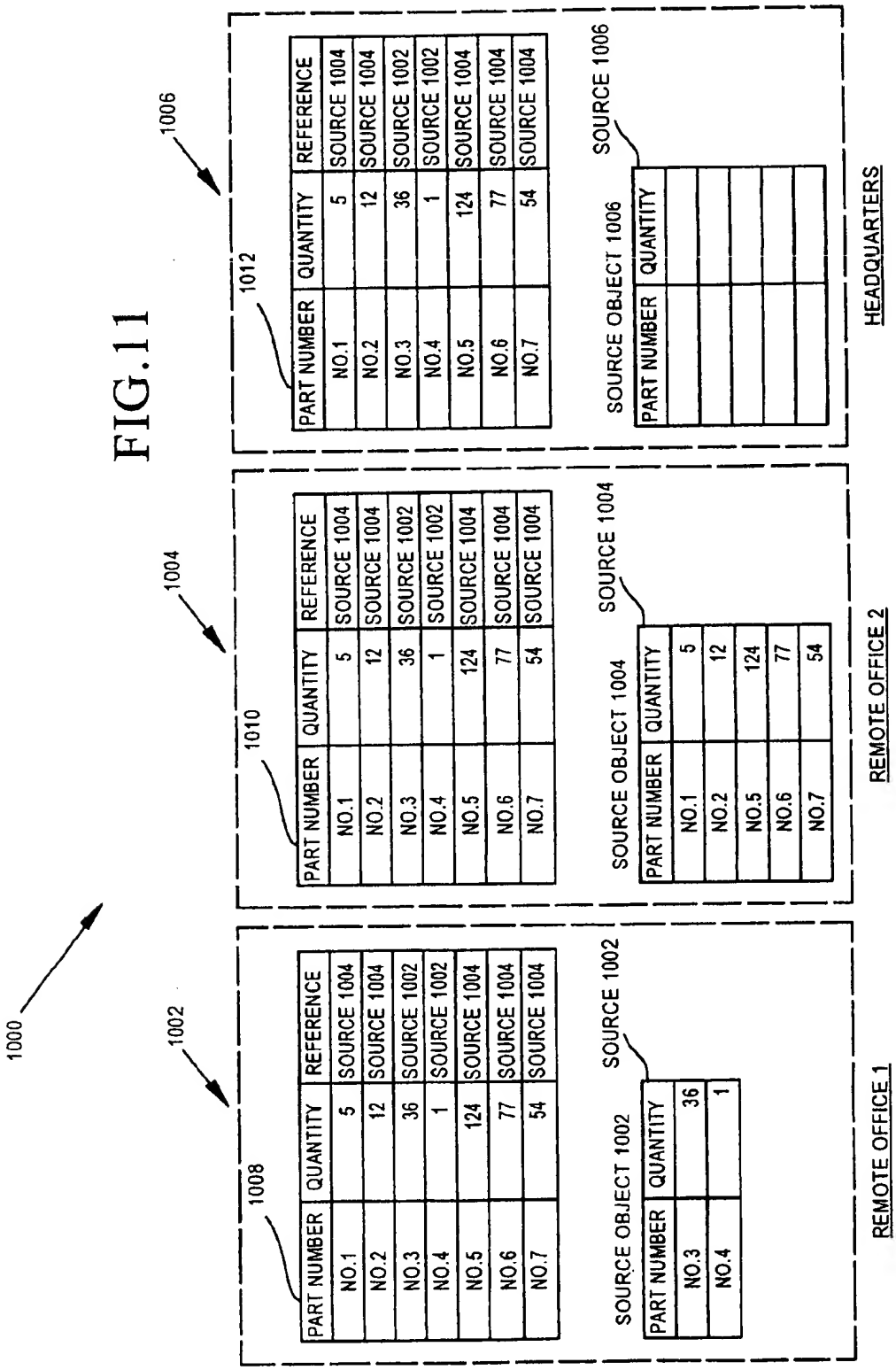

1000
1002
1004
1006
1008
1010
1012
1014
LINKS
PART NUMBER
QUANTITY
REFERENCE
NO.1
NO.2
NO.3
NO.4
NO.5
NO.6
NO.7
5
12
36
1
124
77
54
SOURCE 1004
SOURCE 1004
SOURCE 1002
SOURCE 1002
SOURCE 1004
SOURCE 1004
SOURCE 1004

FIG.10

FIG. 11


1000
1002
1004
1006
1008
1010
1012
REMOTE OFFICE 1
REMOTE OFFICE 2
HEADQUARTERS
SOURCE OBJECT 1002
SOURCE 1002
SOURCE OBJECT 1004
SOURCE 1004
SOURCE OBJECT 1006
SOURCE 1006
PART NUMBER
QUANTITY
REFERENCE
NO.1 5 SOURCE 1004
NO.2 12 SOURCE 1004
NO.3 36 SOURCE 1002
NO.4 1 SOURCE 1002
NO.5 124 SOURCE 1004
NO.6 77 SOURCE 1004
NO.7 54 SOURCE 1004
NO.3 36
NO.4 1
NO.1 5
NO.2 12
NO.5 124
NO.6 77
NO.7 54

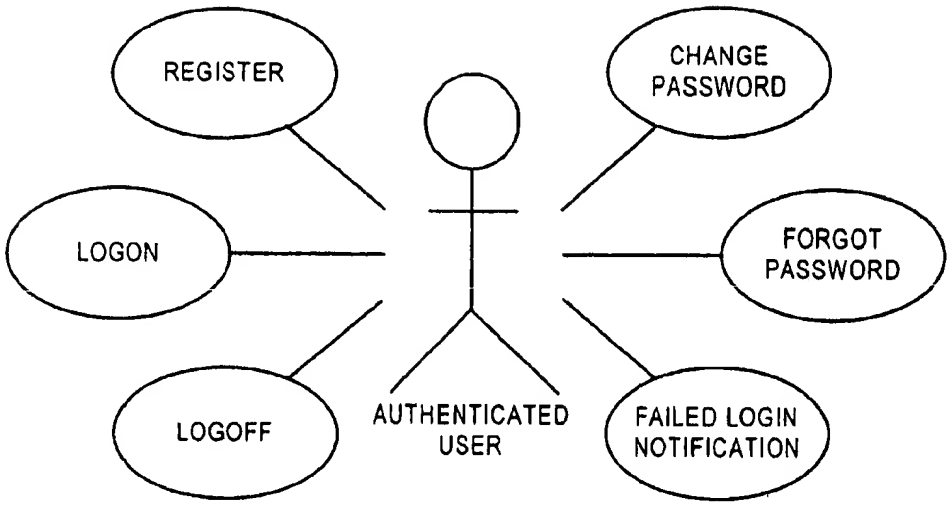

REGISTER
CHANGE PASSWORD
LOGON
FORGOT PASSWORD
LOGOFF
AUTHENTICATED USER
FAILED LOGIN NOTIFICATION

FIG.12

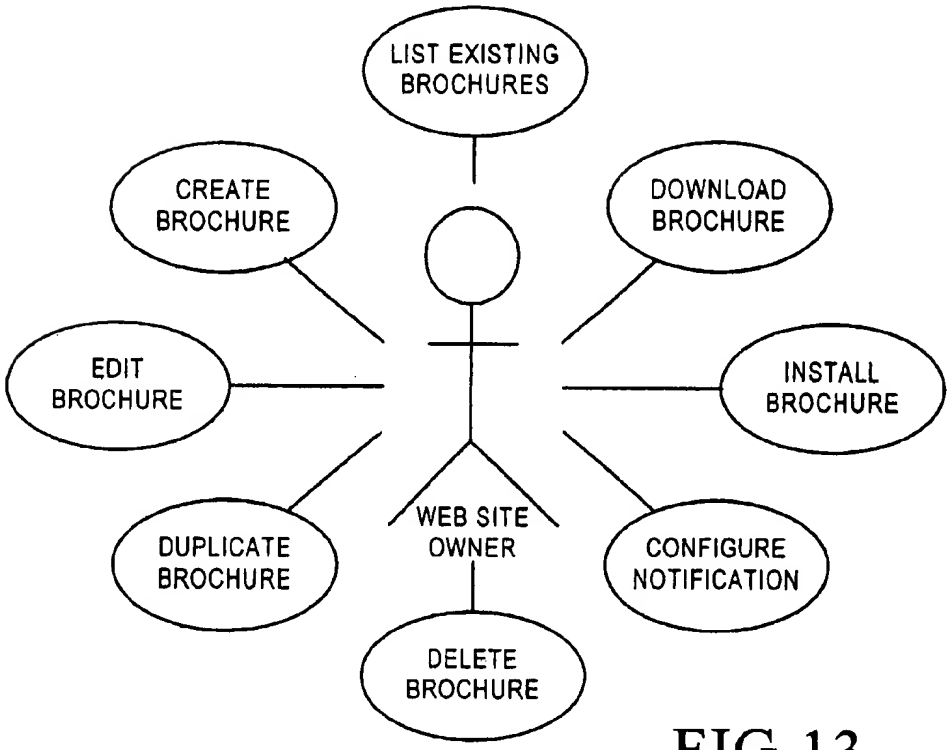

LIST EXISTING BROCHURES
CREATE BROCHURE
DOWNLOAD BROCHURE
EDIT BROCHURE
INSTALL BROCHURE
DUPLICATE BROCHURE
WEB SITE OWNER
CONFIGURE NOTIFICATION
DELETE BROCHURE

FIG.13

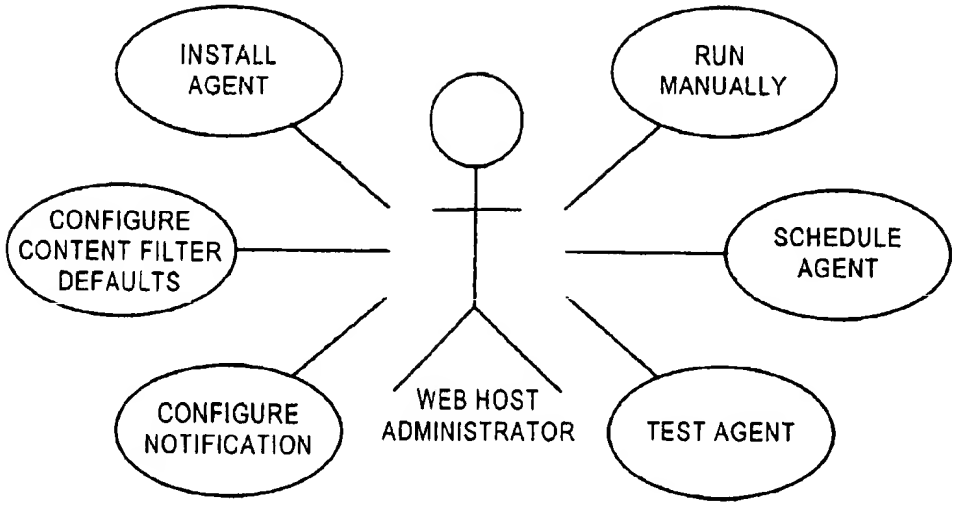

INSTALL AGENT
RUN MANUALLY
CONFIGURE CONTENT FILTER DEFAULTS
SCHEDULE AGENT
CONFIGURE NOTIFICATION
WEB HOST ADMINISTRATOR
TEST AGENT

FIG.14

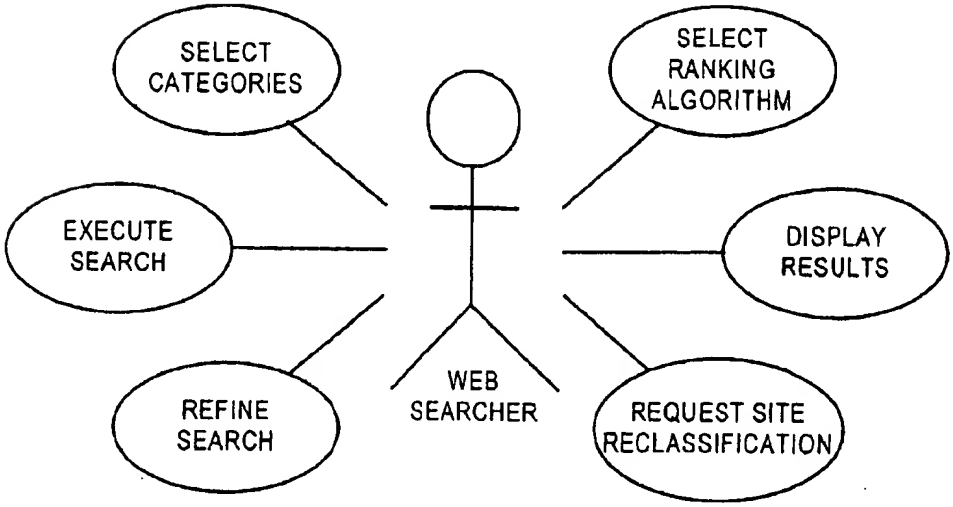

SELECT CATEGORIES
SELECT RANKING ALGORITHM
EXECUTE SEARCH
DISPLAY RESULTS
REFINE SEARCH
WEB SEARCHER
REQUEST SITE RECLASSIFICATION

FIG.15

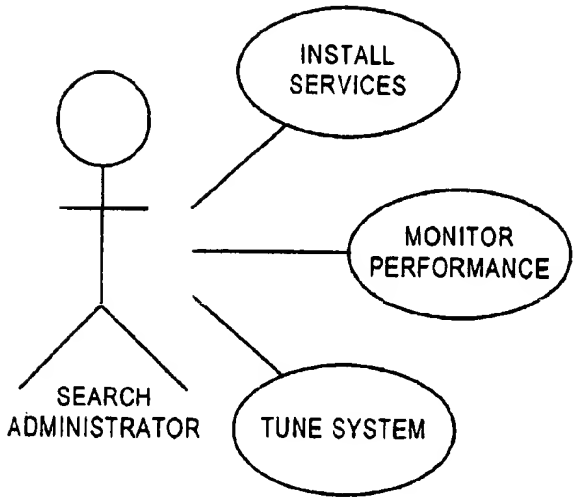

INSTALL SERVICES
MONITOR PERFORMANCE
TUNE SYSTEM
SEARCH ADMINISTRATOR

FIG.16

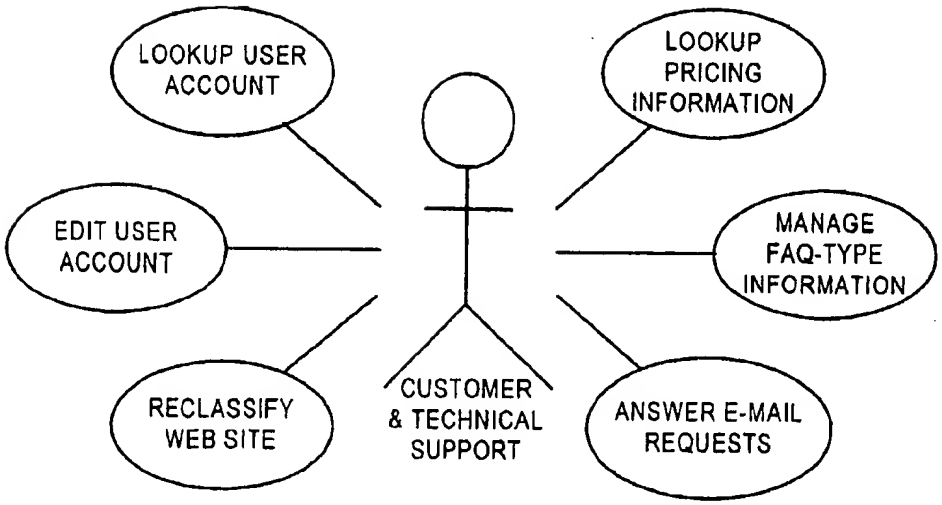

LOOKUP USER ACCOUNT
LOOKUP PRICING INFORMATION
EDIT USER ACCOUNT
MANAGE FAQ-TYPE INFORMATION
RECLASSIFY WEB SITE
CUSTOMER & TECHNICAL SUPPORT
ANSWER E-MAIL REQUESTS

FIG.17

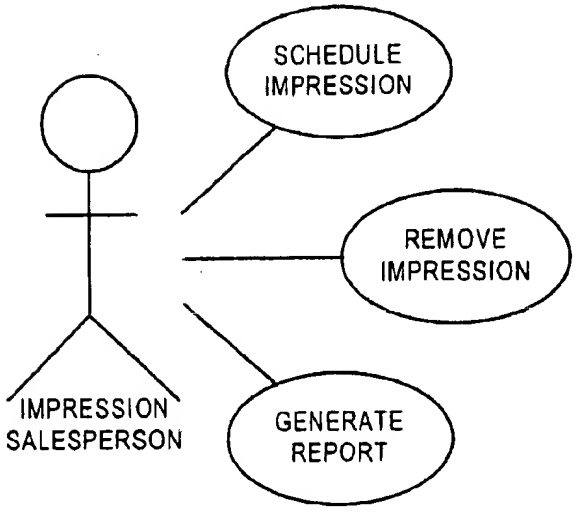

SCHEDULE IMPRESSION
REMOVE IMPRESSION
GENERATE REPORT
IMPRESSION SALESPERSON

FIG.18

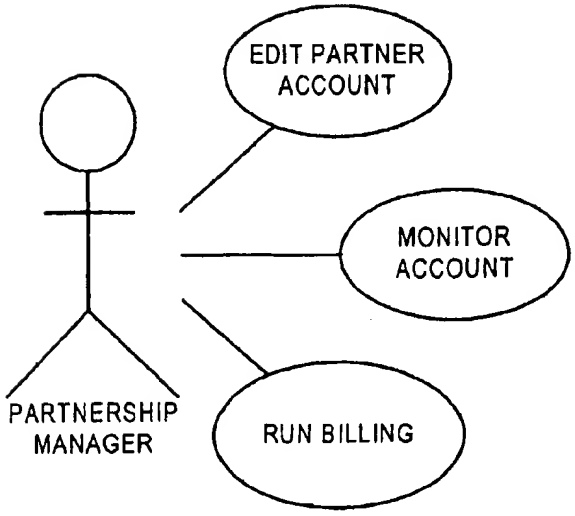

EDIT PARTNER ACCOUNT
MONITOR ACCOUNT
RUN BILLING
PARTNERSHIP MANAGER

FIG.19

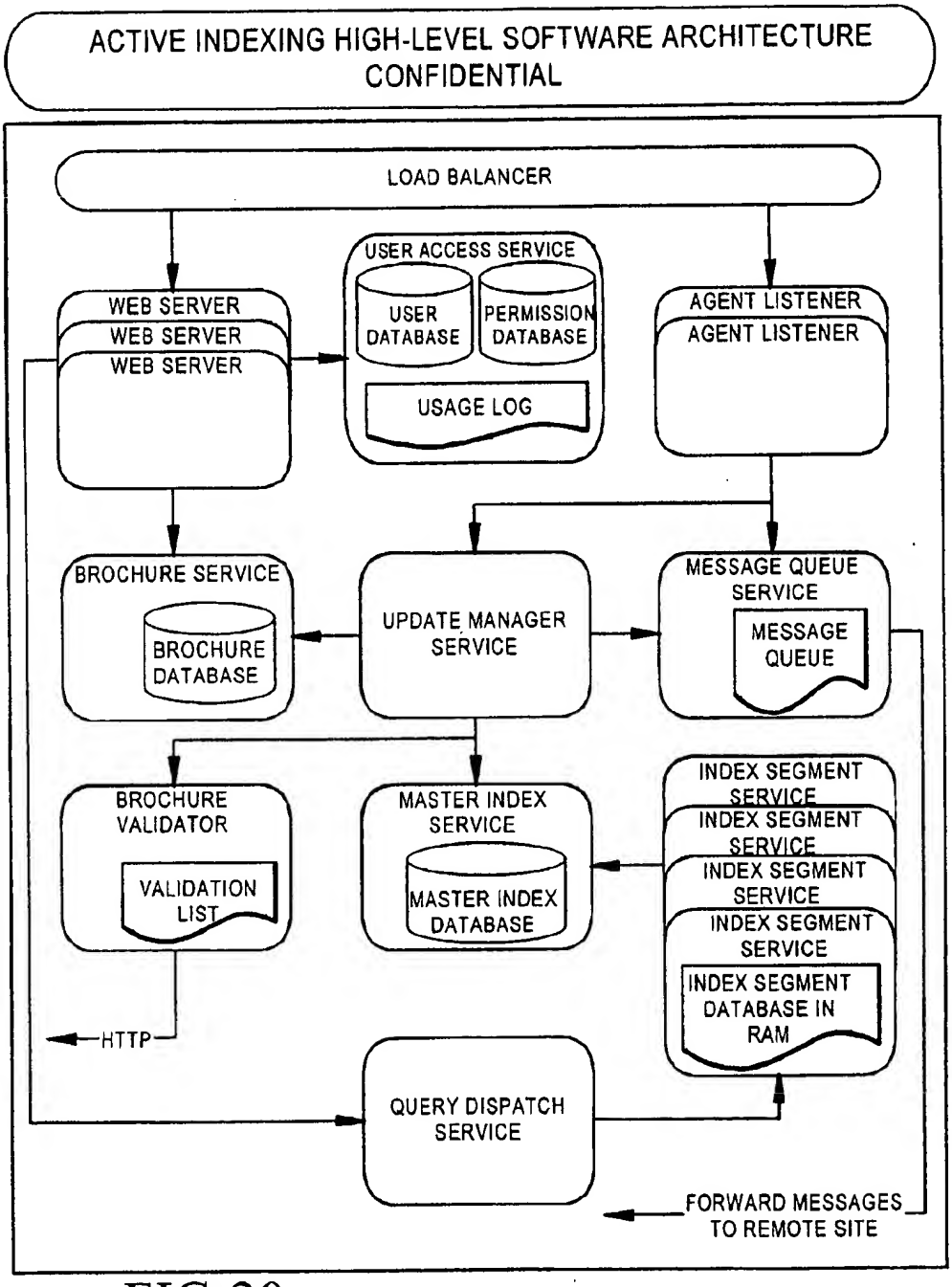

ACTIVE INDEXING HIGH-LEVEL SOFTWARE ARCHITECTURE
CONFIDENTIAL
LOAD BALANCER
WEB SERVER
WEB SERVER
WEB SERVER
USER ACCESS SERVICE
USER DATABASE
PERMISSION DATABASE
USAGE LOG
AGENT LISTENER
AGENT LISTENER
BROCHURE SERVICE
BROCHURE DATABASE
UPDATE MANAGER SERVICE
MESSAGE QUEUE SERVICE
MESSAGE QUEUE
BROCHURE VALIDATOR
VALIDATION LIST
MASTER INDEX SERVICE
MASTER INDEX DATABASE
INDEX SEGMENT SERVICE
INDEX SEGMENT SERVICE
INDEX SEGMENT SERVICE
INDEX SEGMENT SERVICE
INDEX SEGMENT DATABASE IN RAM
HTTP
QUERY DISPATCH SERVICE
FORWARD MESSAGES TO REMOTE SITE

FIG.20

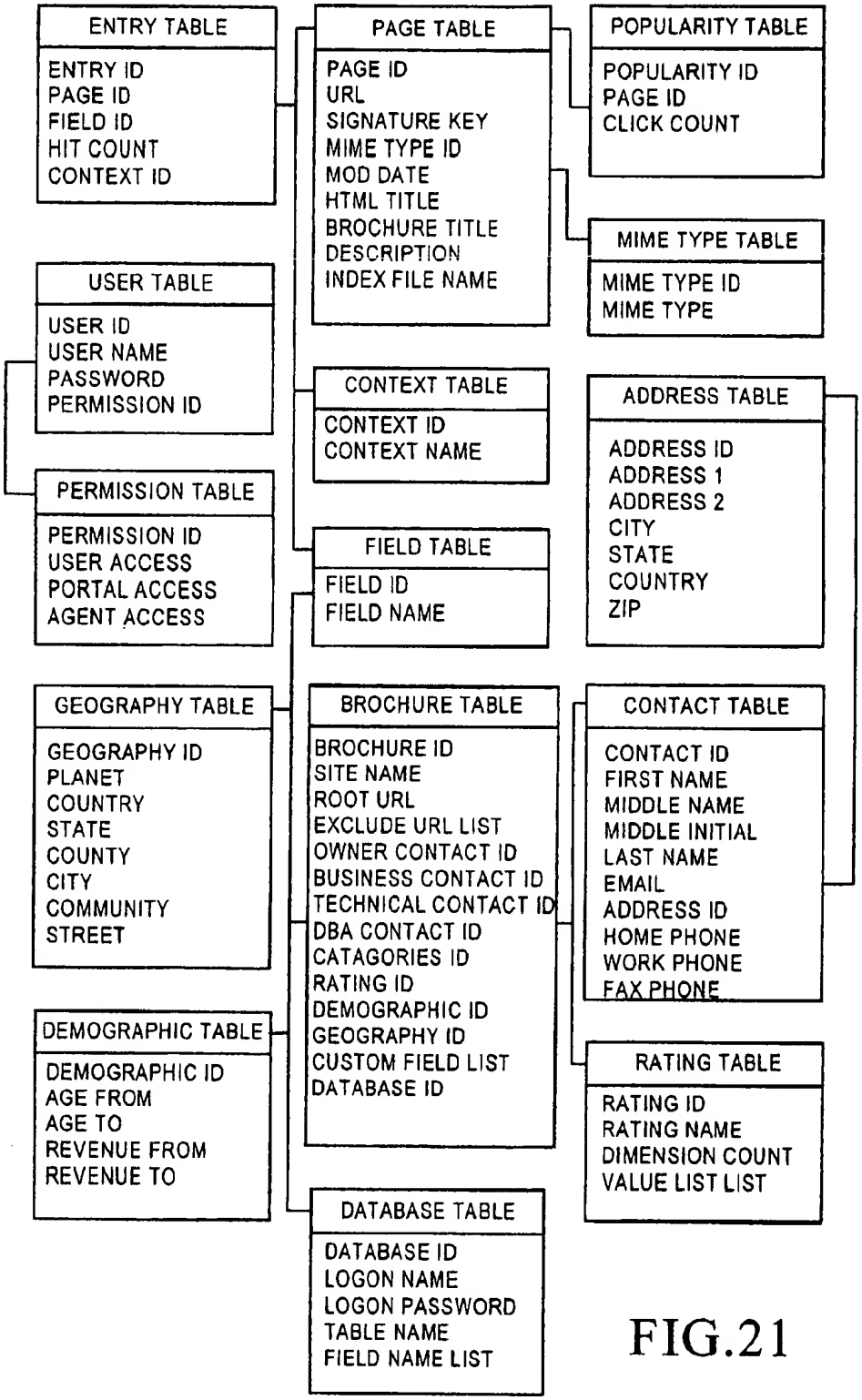

ENTRY TABLE
ENTRY ID
PAGE ID
FIELD ID
HIT COUNT
CONTEXT ID
PAGE TABLE
PAGE ID
URL
SIGNATURE KEY
MIME TYPE ID
MOD DATE
HTML TITLE
BROCHURE TITLE
DESCRIPTION
INDEX FILE NAME
POPULARITY TABLE
POPULARITY ID
PAGE ID
CLICK COUNT
MIME TYPE TABLE
MIME TYPE ID
MIME TYPE
USER TABLE
USER ID
USER NAME
PASSWORD
PERMISSION ID
CONTEXT TABLE
CONTEXT ID
CONTEXT NAME
ADDRESS TABLE
ADDRESS ID
ADDRESS 1
ADDRESS 2
CITY
STATE
COUNTRY
ZIP
PERMISSION TABLE
PERMISSION ID
USER ACCESS
PORTAL ACCESS
AGENT ACCESS
FIELD TABLE
FIELD ID
FIELD NAME
GEOGRAPHY TABLE
GEOGRAPHY ID
PLANET
COUNTRY
STATE
COUNTY
CITY
COMMUNITY
STREET
BROCHURE TABLE
BROCHURE ID
SITE NAME
ROOT URL
EXCLUDE URL LIST
OWNER CONTACT ID
BUSINESS CONTACT ID
TECHNICAL CONTACT ID
DBA CONTACT ID
CATAGORIES ID
RATING ID
DEMOGRAPHIC ID
GEOGRAPHY ID
CUSTOM FIELD LIST
DATABASE ID
CONTACT TABLE
CONTACT ID
FIRST NAME
MIDDLE NAME
MIDDLE INITIAL
LAST NAME
EMAIL
ADDRESS ID
HOME PHONE
WORK PHONE
FAX PHONE
DEMOGRAPHIC TABLE
DEMOGRAPHIC ID
AGE FROM
AGE TO
REVENUE FROM
REVENUE TO
RATING TABLE
RATING ID
RATING NAME
DIMENSION COUNT
VALUE LIST LIST
DATABASE TABLE
DATABASE ID
LOGON NAME
LOGON PASSWORD
TABLE NAME
FIELD NAME LIST

FIG.21

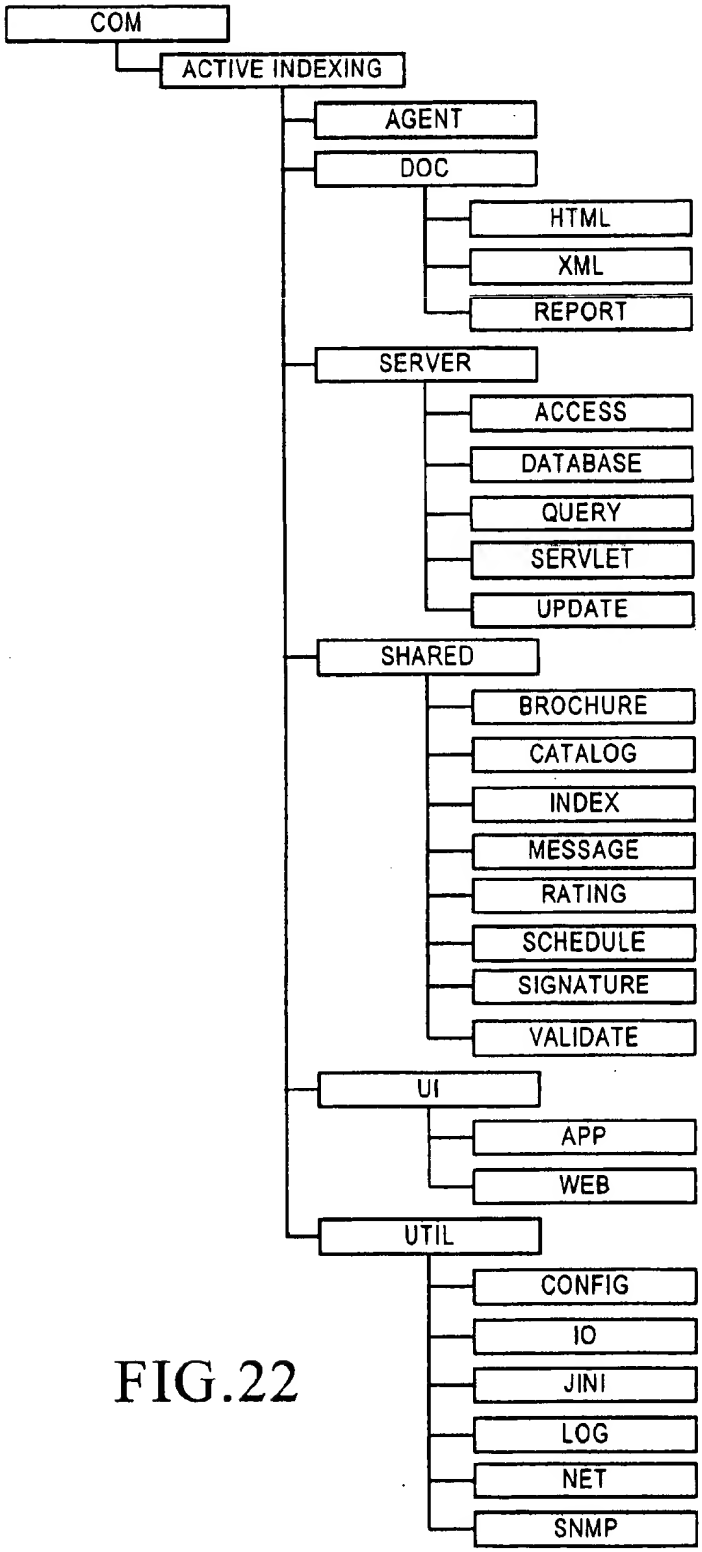

COM
ACTIVE INDEXING
AGENT
DOC
HTML
XML
REPORT
SERVER
ACCESS
DATABASE
QUERY
SERVLET
UPDATE
SHARED
BROCHURE
CATALOG
INDEX
MESSAGE
RATING
SCHEDULE
SIGNATURE
VALIDATE
UI
APP
WEB
UTIL
CONFIG
IO
JINI
LOG
NET
SNMP

FIG.22

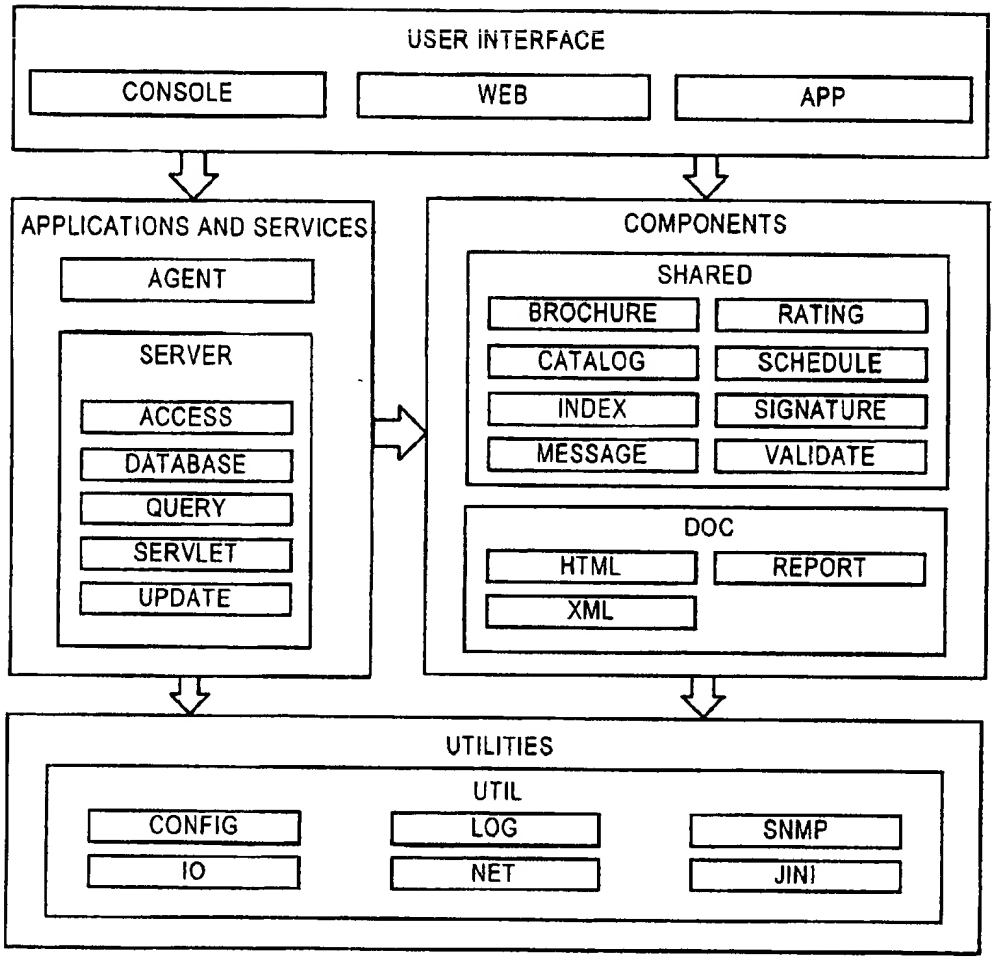

USER INTERFACE
CONSOLE
WEB
APP
APPLICATIONS AND SERVICES
AGENT
SERVER
ACCESS
DATABASE
QUERY
SERVLET
UPDATE
COMPONENTS
SHARED
BROCHURE
RATING
CATALOG
SCHEDULE
INDEX
SIGNATURE
MESSAGE
VALIDATE
DOC
HTML
REPORT
XML
UTILITIES
UTIL
CONFIG
LOG
SNMP
IO
NET
JINI

FIG.23

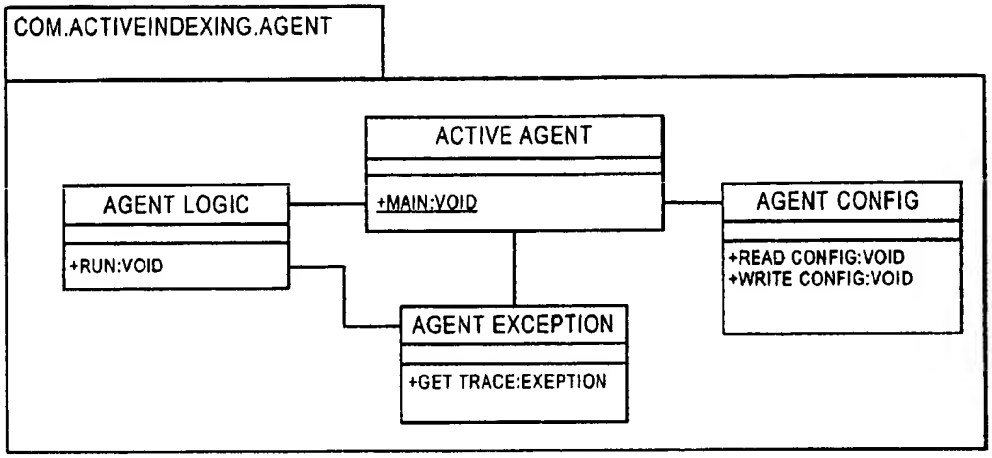

COM.ACTIVEINDEXING.AGENT
ACTIVE AGENT
+MAIN:VOID
AGENT LOGIC
+RUN:VOID
AGENT CONFIG
+READ CONFIG:VOID
+WRITE CONFIG:VOID
AGENT EXCEPTION
+GET TRACE:EXEPTION

FIG.24

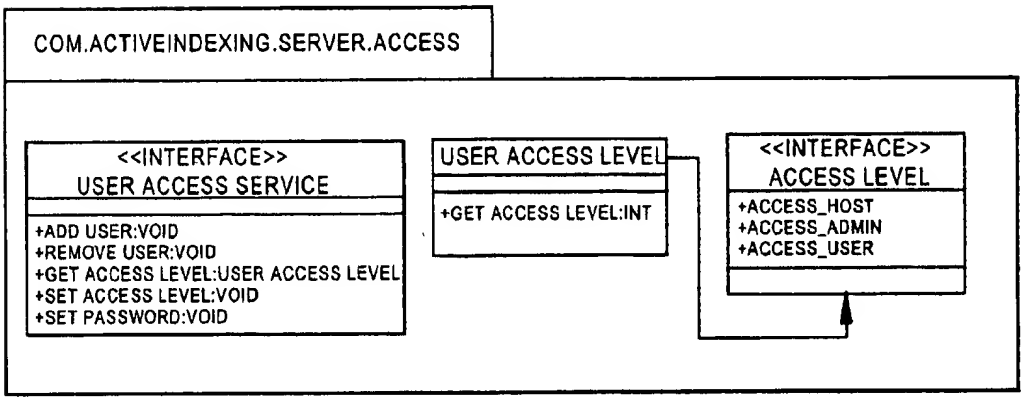

COM.ACTIVEINDEXING.SERVER.ACCESS
<<INTERFACE>>
USER ACCESS SERVICE
+ADD USER:VOID
+REMOVE USER:VOID
+GET ACCESS LEVEL:USER ACCESS LEVEL
+SET ACCESS LEVEL:VOID
+SET PASSWORD:VOID
USER ACCESS LEVEL
+GET ACCESS LEVEL:INT
<<INTERFACE>>
ACCESS LEVEL
+ACCESS_HOST
+ACCESS_ADMIN
+ACCESS_USER

FIG.25

FIG.26

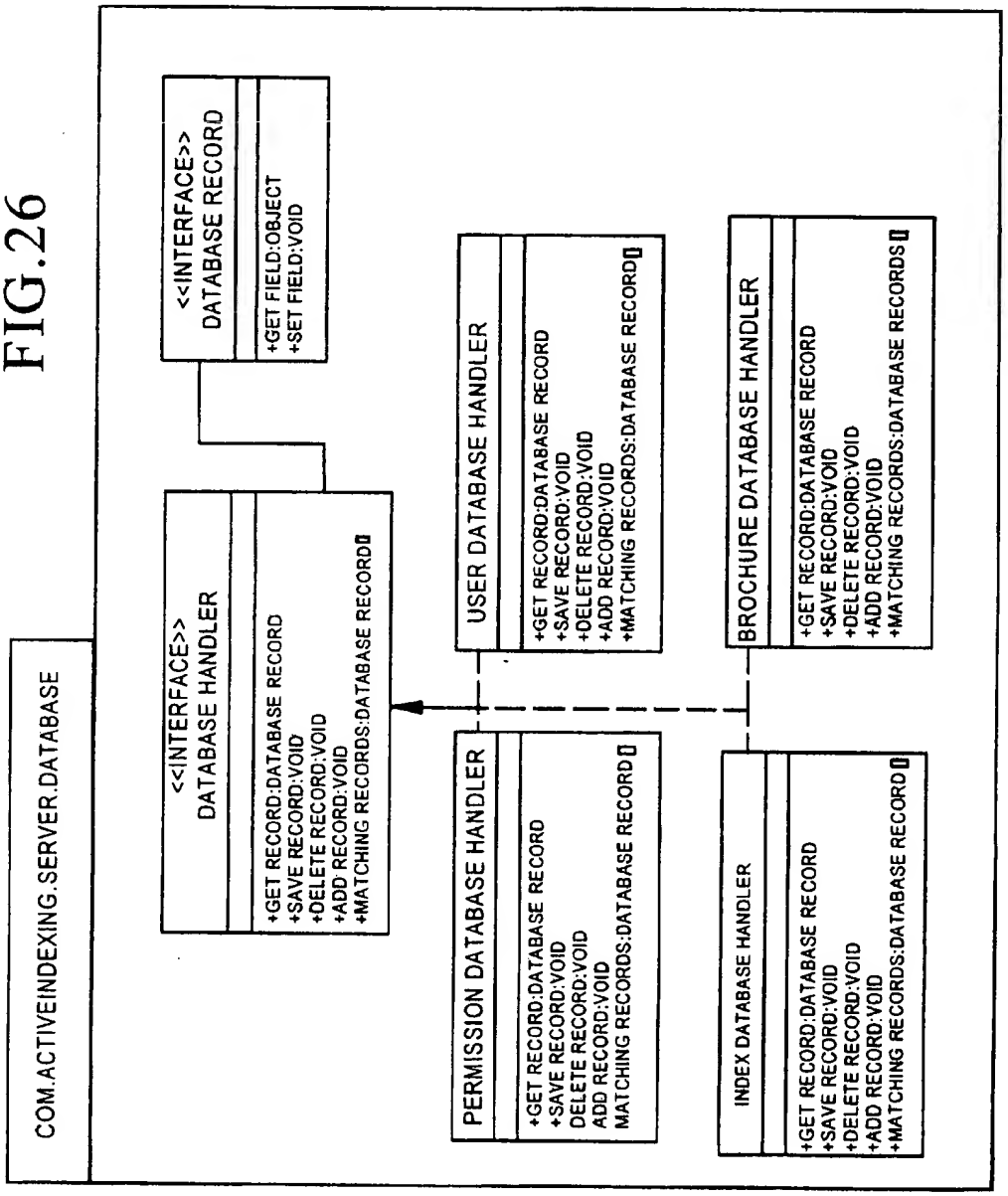

COM.ACTIVEINDEXING.SERVER.DATABASE
<<INTERFACE>>
DATABASE RECORD
+GET FIELD:OBJECT
+SET FIELD:VOID
<<INTERFACE>>
DATABASE HANDLER
+GET RECORD:DATABASE RECORD
+SAVE RECORD:VOID
+DELETE RECORD:VOID
+ADD RECORD:VOID
+MATCHING RECORDS:DATABASE RECORD[]
USER DATABASE HANDLER
+GET RECORD:DATABASE RECORD
+SAVE RECORD:VOID
+DELETE RECORD:VOID
+ADD RECORD:VOID
+MATCHING RECORDS:DATABASE RECORD[]
BROCHURE DATABASE HANDLER
+GET RECORD:DATABASE RECORD
+SAVE RECORD:VOID
+DELETE RECORD:VOID
+ADD RECORD:VOID
+MATCHING RECORDS:DATABASE RECORDS[]
PERMISSION DATABASE HANDLER
+GET RECORD:DATABASE RECORD
+SAVE RECORD:VOID
DELETE RECORD:VOID
ADD RECORD:VOID
MATCHING RECORDS:DATABASE RECORD[]
INDEX DATABASE HANDLER
+GET RECORD:DATABASE RECORD
+SAVE RECORD:VOID
+DELETE RECORD:VOID
+ADD RECORD:VOID
+MATCHING RECORDS:DATABASE RECORD[]

FIG.27

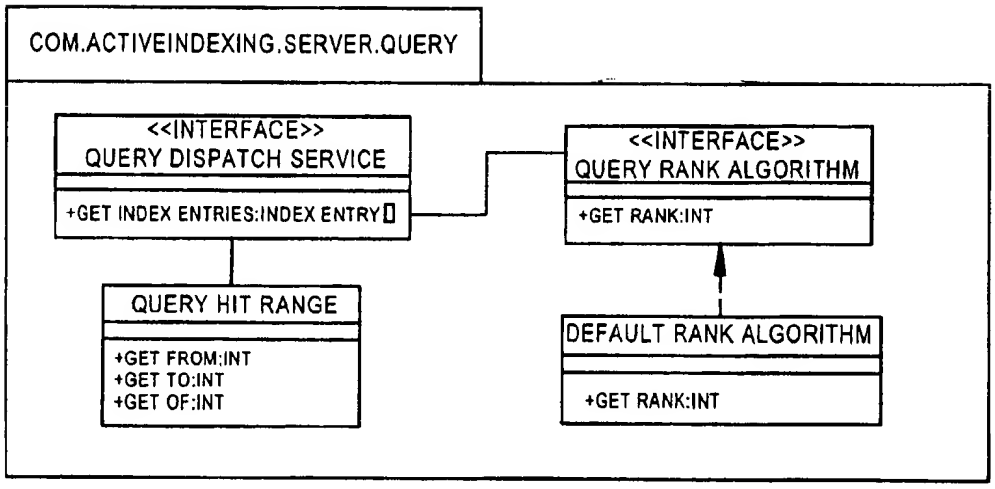

COM.ACTIVEINDEXING.SERVER.QUERY
<<INTERFACE>>
QUERY DISPATCH SERVICE
+GET INDEX ENTRIES:INDEX ENTRY[]
<<INTERFACE>>
QUERY RANK ALGORITHM
+GET RANK:INT
QUERY HIT RANGE
+GET FROM:INT
+GET TO:INT
+GET OF:INT
DEFAULT RANK ALGORITHM
+GET RANK:INT

FIG.28

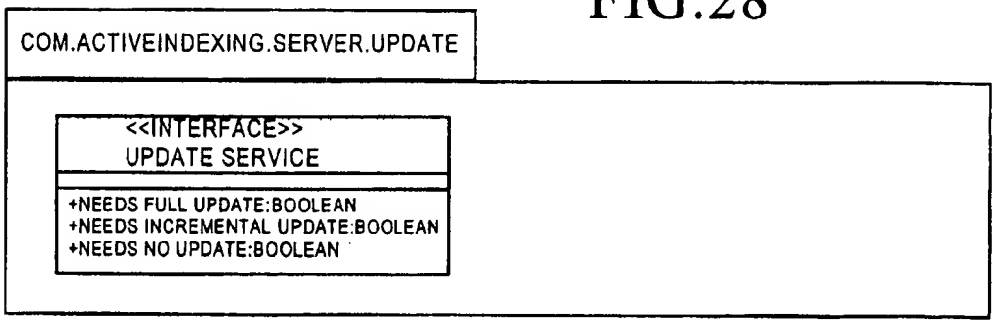

COM.ACTIVEINDEXING.SERVER.UPDATE
<<INTERFACE>>
UPDATE SERVICE
+NEEDS FULL UPDATE:BOOLEAN
+NEEDS INCREMENTAL UPDATE:BOOLEAN
+NEEDS NO UPDATE:BOOLEAN

FIG.29

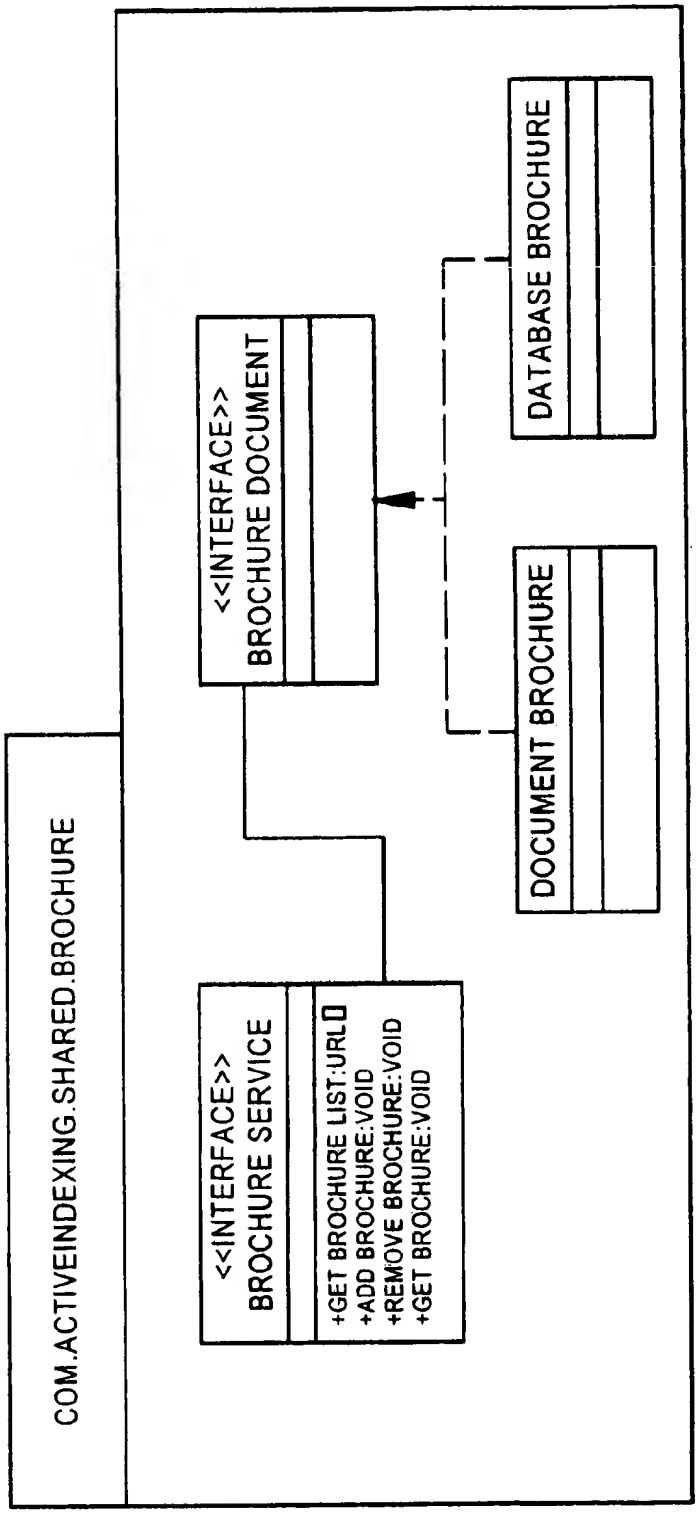

COM.ACTIVEINDEXING.SHARED.BROCHURE
<<INTERFACE>>
BROCHURE SERVICE
+GET BROCHURE LIST:URL[]
+ADD BROCHURE:VOID
+REMOVE BROCHURE:VOID
+GET BROCHURE:VOID
<<INTERFACE>>
BROCHURE DOCUMENT
DOCUMENT BROCHURE
DATABASE BROCHURE

FIG.30

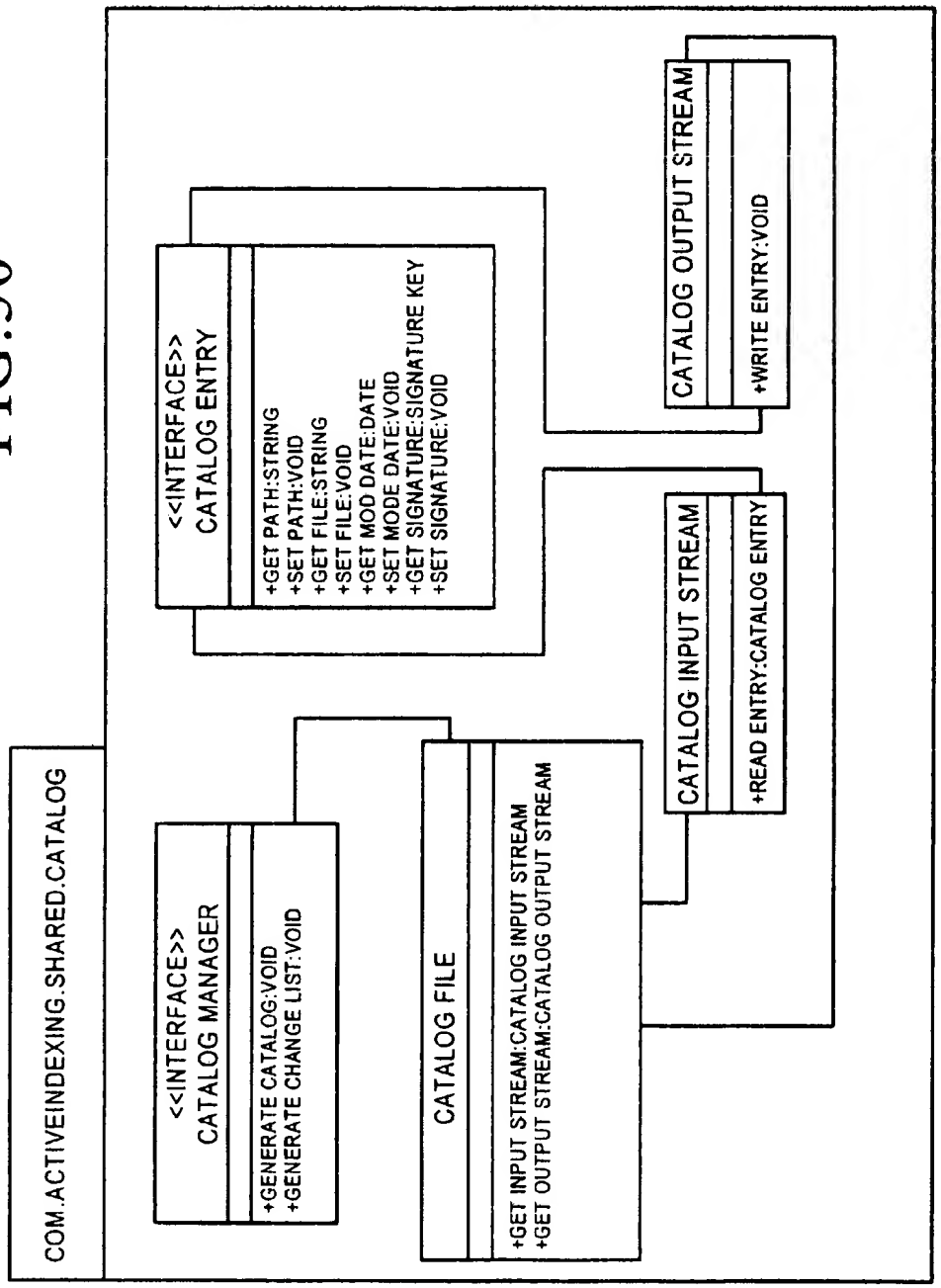
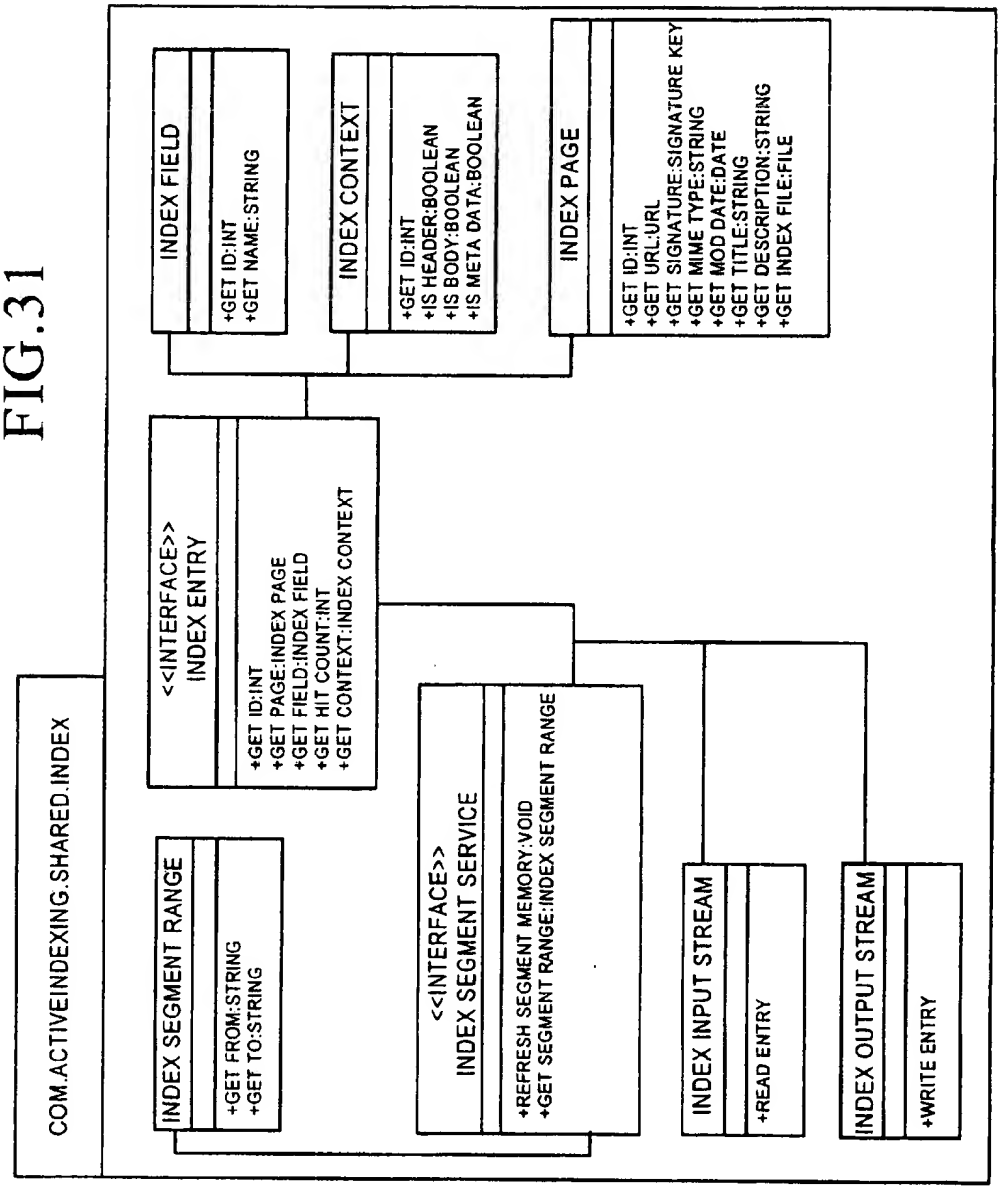

COM.ACTIVEINDEXING.SHARED.CATALOG
<<INTERFACE>>
CATALOG MANAGER
+GENERATE CATALOG:VOID
+GENERATE CHANGE LIST:VOID
<<INTERFACE>>
CATALOG ENTRY
+GET PATH:STRING
+SET PATH:VOID
+GET FILE:STRING
+SET FILE:VOID
+GET MOD DATE:DATE
+SET MODE DATE:VOID
+GET SIGNATURE:SIGNATURE KEY
+SET SIGNATURE:VOID
CATALOG FILE
+GET INPUT STREAM:CATALOG INPUT STREAM
+GET OUTPUT STREAM:CATALOG OUTPUT STREAM
CATALOG INPUT STREAM
+READ ENTRY:CATALOG ENTRY
CATALOG OUTPUT STREAM
+WRITE ENTRY:VOID

FIG.31


COM.ACTIVEINDEXING.SHARED.INDEX
INDEX FIELD
+GET ID:INT
+GET NAME:STRING
INDEX CONTEXT
+GET ID:INT
+IS HEADER:BOOLEAN
+IS BODY:BOOLEAN
+IS META DATA:BOOLEAN
INDEX PAGE
+GET ID:INT
+GET URL:URL
+GET SIGNATURE:SIGNATURE KEY
+GET MIME TYPE:STRING
+GET MOD DATE:DATE
+GET TITLE:STRING
+GET DESCRIPTION:STRING
+GET INDEX FILE:FILE
<<INTERFACE>>
INDEX ENTRY
+GET ID:INT
+GET PAGE:INDEX PAGE
+GET FIELD:INDEX FIELD
+GET HIT COUNT:INT
+GET CONTEXT:INDEX CONTEXT
INDEX SEGMENT RANGE
+GET FROM:STRING
+GET TO:STRING
<<INTERFACE>>
INDEX SEGMENT SERVICE
+REFRESH SEGMENT MEMORY:VOID
+GET SEGMENT RANGE:INDEX SEGMENT RANGE
INDEX INPUT STREAM
+READ ENTRY
INDEX OUTPUT STREAM
+WRITE ENTRY

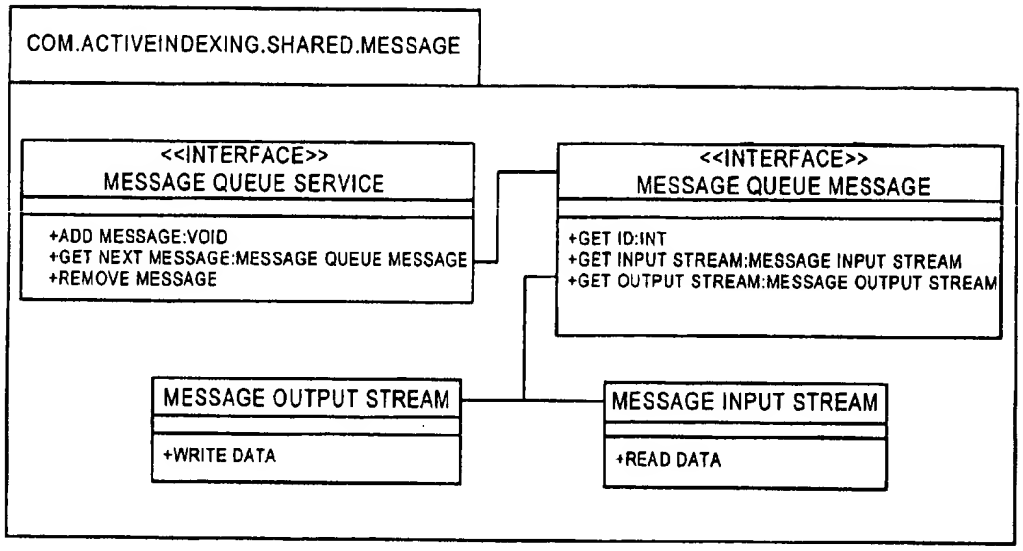
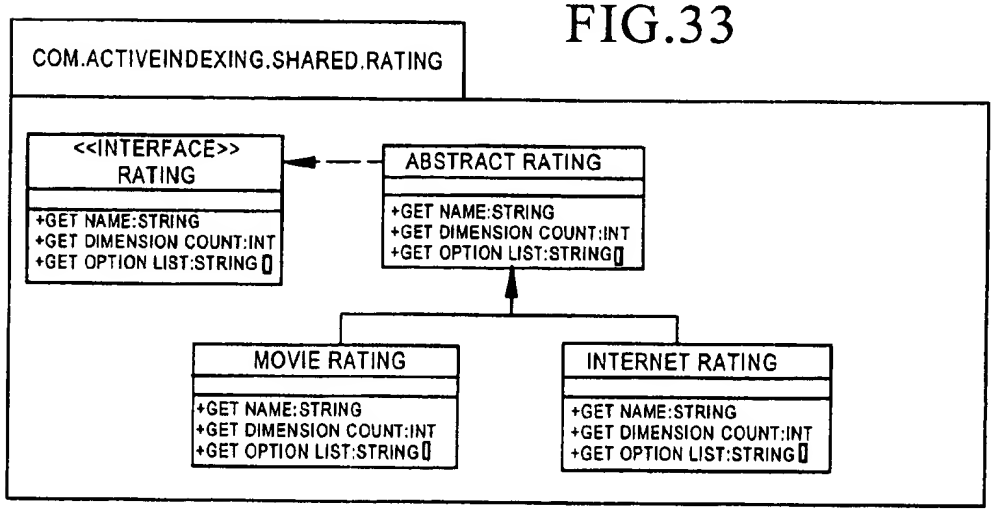

COM.ACTIVEINDEXING.SHARED.MESSAGE
<<INTERFACE>>
MESSAGE QUEUE SERVICE
+ADD MESSAGE:VOID
+GET NEXT MESSAGE:MESSAGE QUEUE MESSAGE
+REMOVE MESSAGE
<<INTERFACE>>
MESSAGE QUEUE MESSAGE
+GET ID:INT
+GET INPUT STREAM:MESSAGE INPUT STREAM
+GET OUTPUT STREAM:MESSAGE OUTPUT STREAM
MESSAGE OUTPUT STREAM
+WRITE DATA
MESSAGE INPUT STREAM
+READ DATA

FIG.32

FIG.33


COM.ACTIVEINDEXING.SHARED.RATING
<<INTERFACE>>
RATING
+GET NAME:STRING
+GET DIMENSION COUNT:INT
+GET OPTION LIST:STRING[]
ABSTRACT RATING
+GET NAME:STRING
+GET DIMENSION COUNT:INT
+GET OPTION LIST:STRING[]
MOVIE RATING
+GET NAME:STRING
+GET DIMENSION COUNT:INT
+GET OPTION LIST:STRING[]
INTERNET RATING
+GET NAME:STRING
+GET DIMENSION COUNT:INT
+GET OPTION LIST:STRING[]

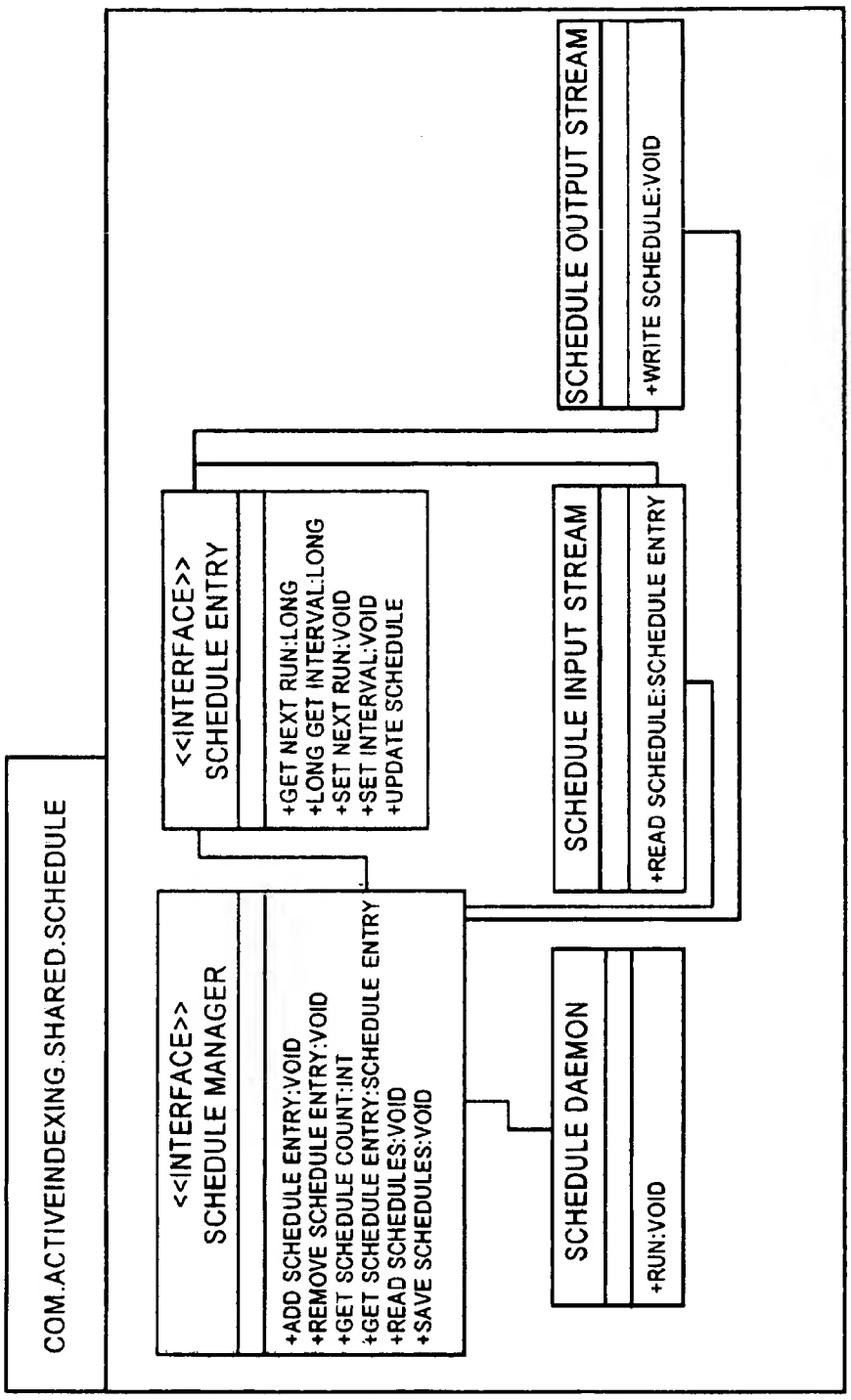

COM.ACTIVEINDEXING.SHARED.SCHEDULE
<<INTERFACE>>
SCHEDULE MANAGER
+ADD SCHEDULE ENTRY:VOID
+REMOVE SCHEDULE ENTRY:VOID
+GET SCHEDULE COUNT:INT
+GET SCHEDULE ENTRY:SCHEDULE ENTRY
+READ SCHEDULES:VOID
+SAVE SCHEDULES:VOID
<<INTERFACE>>
SCHEDULE ENTRY
+GET NEXT RUN:LONG
+LONG GET INTERVAL:LONG
+SET NEXT RUN:VOID
+SET INTERVAL:VOID
+UPDATE SCHEDULE
SCHEDULE DAEMON
+RUN:VOID
SCHEDULE INPUT STREAM
+READ SCHEDULE:SCHEDULE ENTRY
SCHEDULE OUTPUT STREAM
+WRITE SCHEDULE:VOID

FIG.34

FIG.35

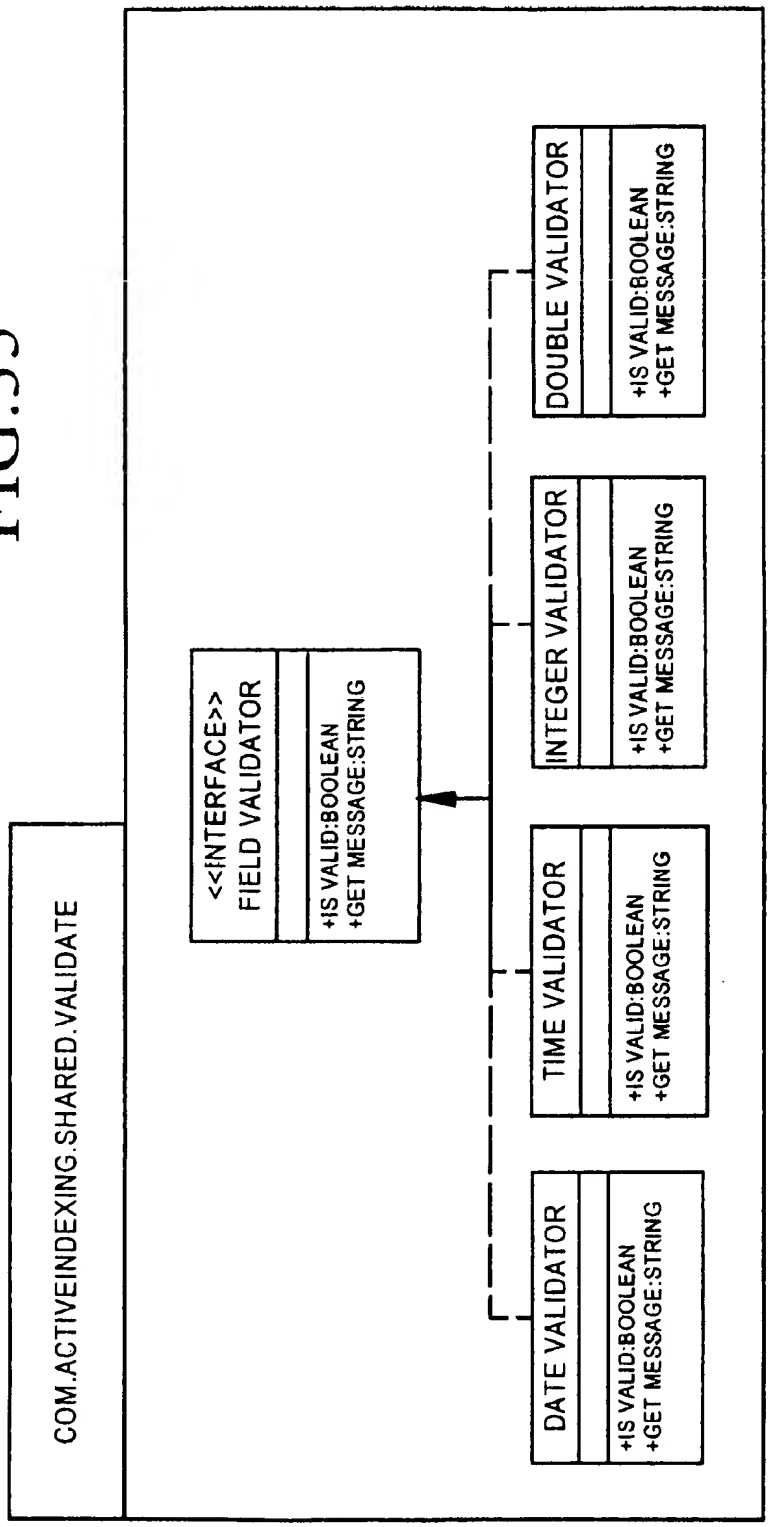
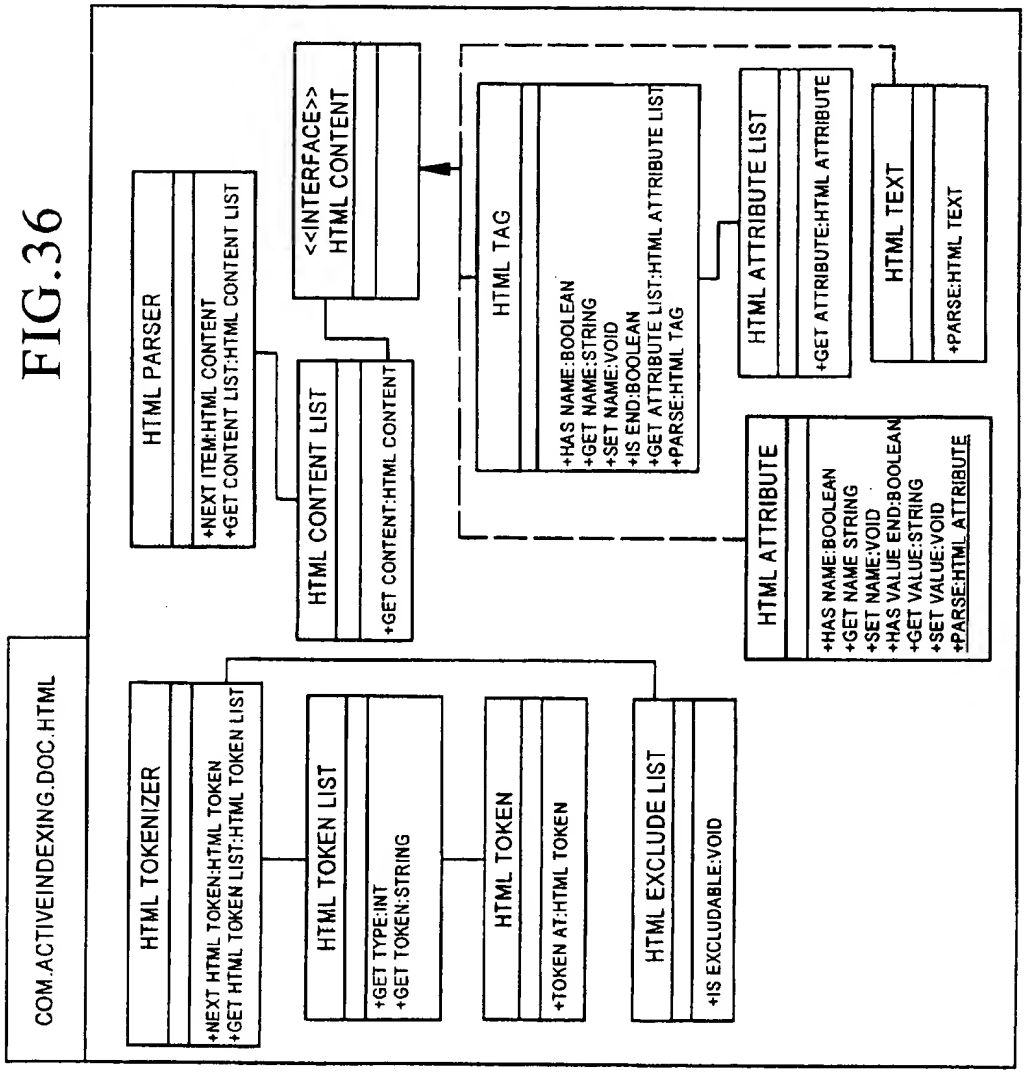

COM.ACTIVEINDEXING.SHARED.VALIDATE
<<INTERFACE>>
FIELD VALIDATOR
+IS VALID:BOOLEAN
+GET MESSAGE:STRING
DATE VALIDATOR
+IS VALID:BOOLEAN
+GET MESSAGE:STRING
TIME VALIDATOR
+IS VALID:BOOLEAN
+GET MESSAGE:STRING
INTEGER VALIDATOR
+IS VALID:BOOLEAN
+GET MESSAGE:STRING
DOUBLE VALIDATOR
+IS VALID:BOOLEAN
+GET MESSAGE:STRING

FIG.36


COM.ACTIVEINDEXING.DOC.HTML
HTML PARSER
+NEXT ITEM:HTML CONTENT
+GET CONTENT LIST:HTML CONTENT LIST
<<INTERFACE>>
HTML CONTENT
HTML CONTENT LIST
+GET CONTENT:HTML CONTENT
HTML TAG
+HAS NAME:BOOLEAN
+GET NAME:STRING
+SET NAME:VOID
+IS END:BOOLEAN
+GET ATTRIBUTE LIST:HTML ATTRIBUTE LIST
+PARSE:HTML TAG
HTML ATTRIBUTE LIST
+GET ATTRIBUTE:HTML ATTRIBUTE
HTML TEXT
+PARSE:HTML TEXT
HTML ATTRIBUTE
+HAS NAME:BOOLEAN
+GET NAME STRING
+SET NAME:VOID
+HAS VALUE END:BOOLEAN
+GET VALUE:STRING
+SET VALUE:VOID
+PARSE:HTML ATTRIBUTE
HTML TOKENIZER
+NEXT HTML TOKEN:HTML TOKEN
+GET HTML TOKEN LIST:HTML TOKEN LIST
HTML TOKEN LIST
+GET TYPE:INT
+GET TOKEN:STRING
HTML TOKEN
+TOKEN AT:HTML TOKEN
HTML EXCLUDE LIST
+IS EXCLUDABLE:VOID

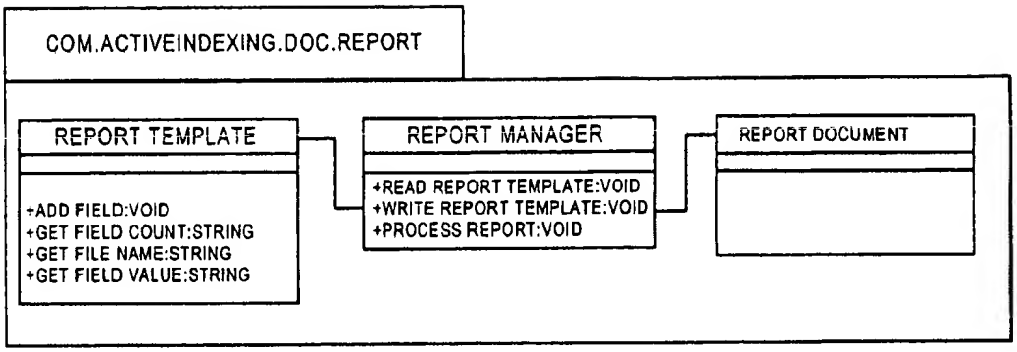

COM.ACTIVEINDEXING.DOC.REPORT
REPORT TEMPLATE
+ADD FIELD:VOID
+GET FIELD COUNT:STRING
+GET FILE NAME:STRING
+GET FIELD VALUE:STRING
REPORT MANAGER
+READ REPORT TEMPLATE:VOID
+WRITE REPORT TEMPLATE:VOID
+PROCESS REPORT:VOID
REPORT DOCUMENT

FIG.37

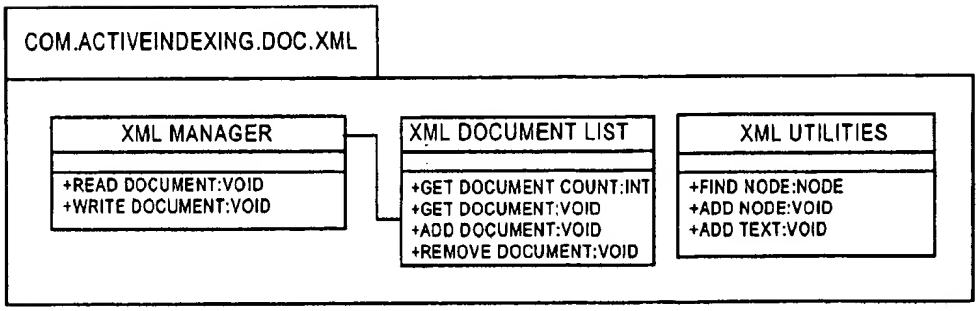

COM.ACTIVEINDEXING.DOC.XML
XML MANAGER
+READ DOCUMENT:VOID
+WRITE DOCUMENT:VOID
XML DOCUMENT LIST
+GET DOCUMENT COUNT:INT
+GET DOCUMENT:VOID
+ADD DOCUMENT:VOID
+REMOVE DOCUMENT:VOID
XML UTILITIES
+FIND NODE:NODE
+ADD NODE:VOID
+ADD TEXT:VOID

FIG.38

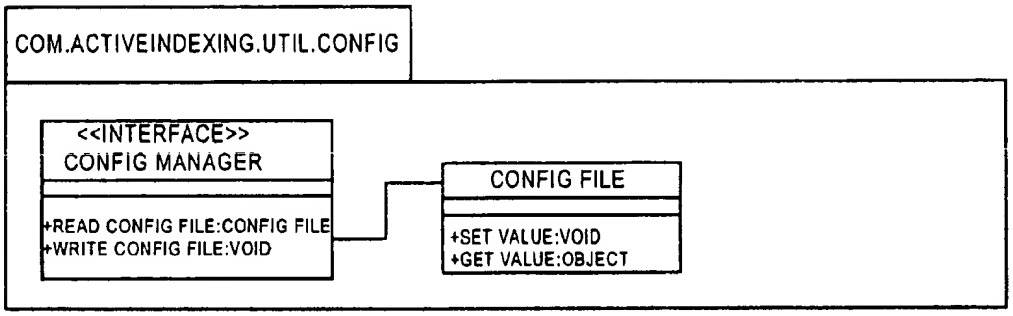
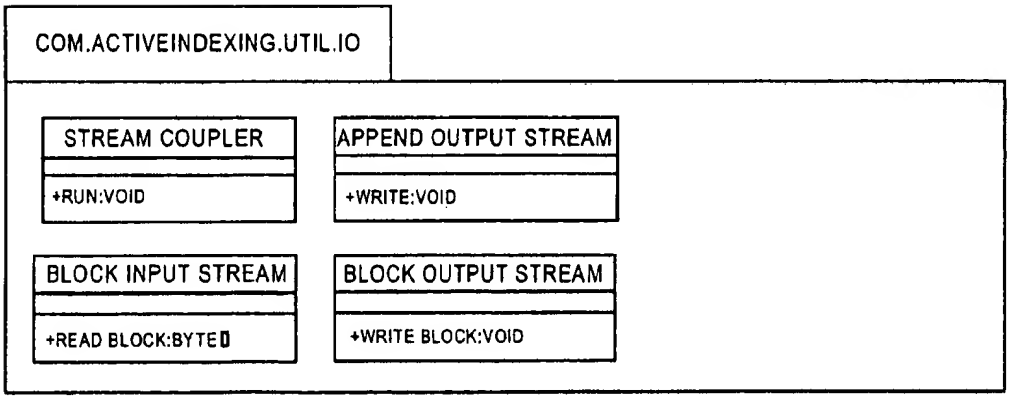

COM.ACTIVEINDEXING.UTIL.CONFIG
<<INTERFACE>>
CONFIG MANAGER
+READ CONFIG FILE:CONFIG FILE
+WRITE CONFIG FILE:VOID
CONFIG FILE
+SET VALUE:VOID
+GET VALUE:OBJECT

FIG.39

FIG.40


COM.ACTIVEINDEXING.UTIL.IO
STREAM COUPLER
+RUN:VOID
APPEND OUTPUT STREAM
+WRITE:VOID
BLOCK INPUT STREAM
+READ BLOCK:BYTE[]
BLOCK OUTPUT STREAM
+WRITE BLOCK:VOID

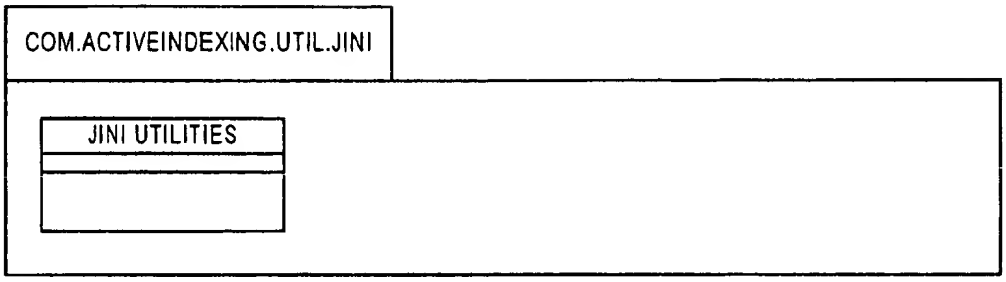
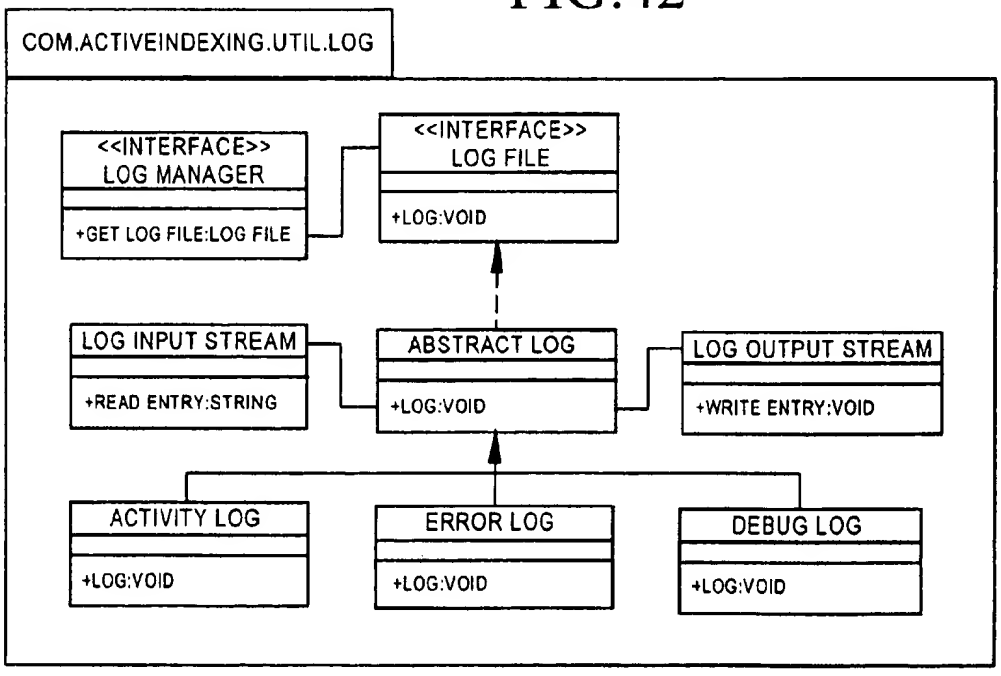

COM.ACTIVEINDEXING.UTIL.JINI
JINI UTILITIES

FIG.41

FIG.42


COM.ACTIVEINDEXING.UTIL.LOG
<<INTERFACE>>
LOG MANAGER
+GET LOG FILE:LOG FILE
<<INTERFACE>>
LOG FILE
+LOG:VOID
LOG INPUT STREAM
+READ ENTRY:STRING
ABSTRACT LOG
+LOG:VOID
LOG OUTPUT STREAM
+WRITE ENTRY:VOID
ACTIVITY LOG
+LOG:VOID
ERROR LOG
+LOG:VOID
DEBUG LOG
+LOG:VOID

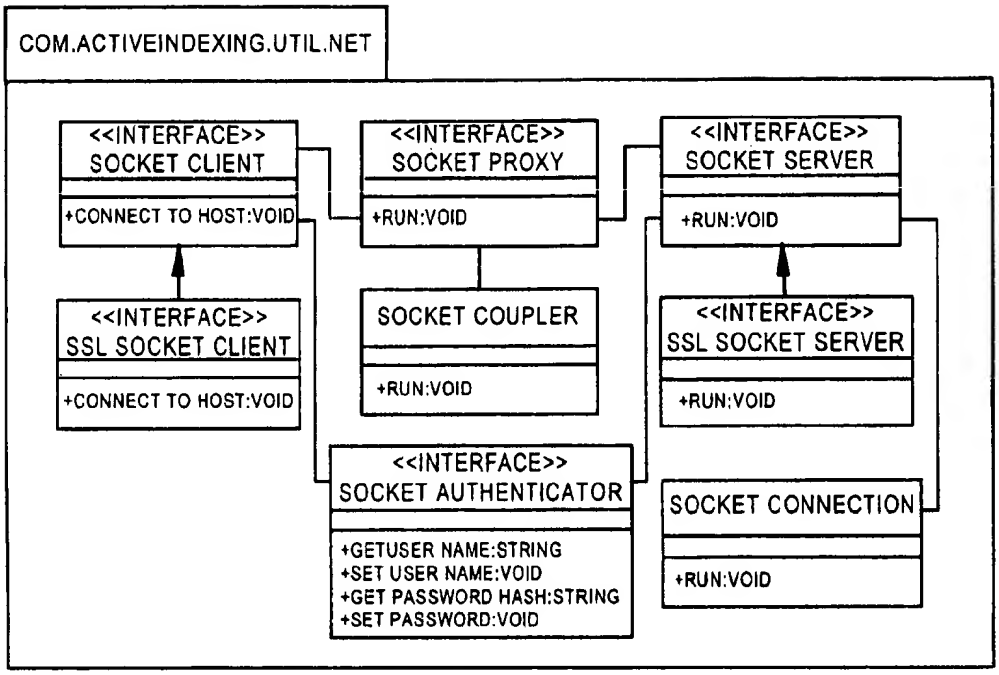
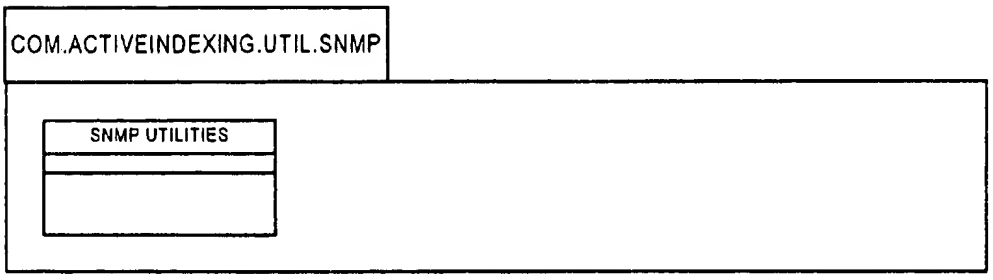

COM.ACTIVEINDEXING.UTIL.NET
<<INTERFACE>>
SOCKET CLIENT
+CONNECT TO HOST:VOID
<<INTERFACE>>
SOCKET PROXY
+RUN:VOID
<<INTERFACE>>
SOCKET SERVER
+RUN:VOID
<<INTERFACE>>
SSL SOCKET CLIENT
+CONNECT TO HOST:VOID
SOCKET COUPLER
+RUN:VOID
<<INTERFACE>>
SSL SOCKET SERVER
+RUN:VOID
<<INTERFACE>>
SOCKET AUTHENTICATOR
+GETUSER NAME:STRING
+SET USER NAME:VOID
+GET PASSWORD HASH:STRING
+SET PASSWORD:VOID
SOCKET CONNECTION
+RUN:VOID

FIG.43

FIG.44


COM.ACTIVEINDEXING.UTIL.SNMP
SNMP UTILITIES

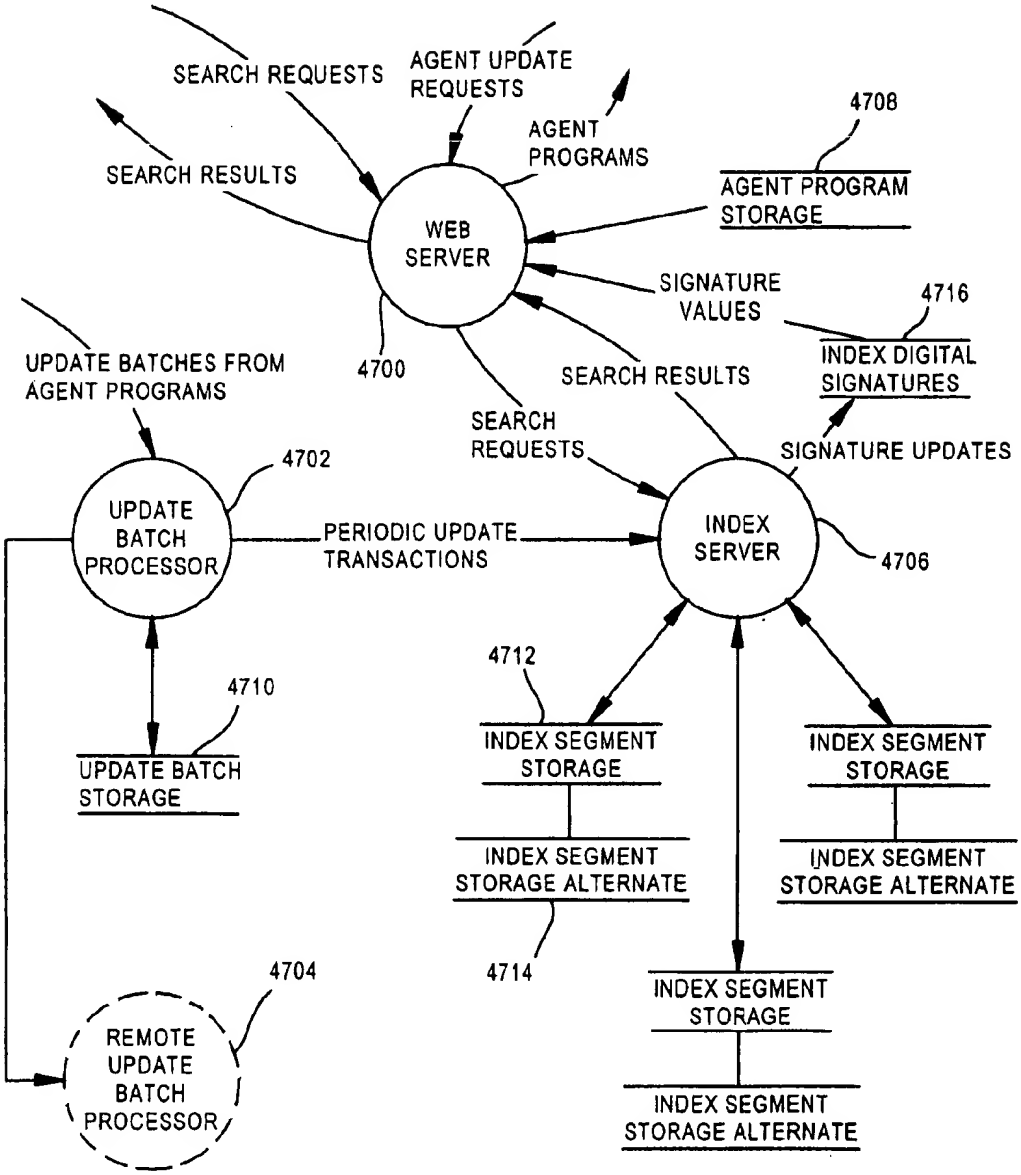

SEARCH REQUESTS
AGENT UPDATE REQUESTS
AGENT PROGRAMS
4708
SEARCH RESULTS
AGENT PROGRAM STORAGE
WEB SERVER
SIGNATURE VALUES
4716
INDEX DIGITAL SIGNATURES
UPDATE BATCHES FROM AGENT PROGRAMS
4700
SEARCH RESULTS
SEARCH REQUESTS
SIGNATURE UPDATES
4702
UPDATE BATCH PROCESSOR
PERIODIC UPDATE TRANSACTIONS
INDEX SERVER
4706
4712
4710
INDEX SEGMENT STORAGE
INDEX SEGMENT STORAGE
UPDATE BATCH STORAGE
INDEX SEGMENT STORAGE ALTERNATE
INDEX SEGMENT STORAGE ALTERNATE
4714
4704
INDEX SEGMENT STORAGE
REMOTE UPDATE BATCH PROCESSOR
INDEX SEGMENT STORAGE ALTERNATE

FIG.45

PEER-TO-PEER AUTOMATED ANONYMOUS ASYNCHRONOUS FILE SHARING

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation in part of U.S. patent applications Ser. Nos. 60/219,983 filed on Jul. 21, 2000, 09/419,405 filed on Oct. 14, 1999, now U.S. Pat. No. 6,516,337, and 09/575,971 filed on May 23, 2000.

BACKGROUND OF THE INVENTION

A number of file discovery and sharing programs have become very popular for use across networks, especially those programs which permit the sharing of multimedia content. Users connect to a central directory service and upload a list of files that they currently have on their local system which may be requested by other participants in the directory service. To retrieve files, users send a request for a file to the central directory service which then connects the requesting user to another user's computer containing that file which computer is also currently online. The most popular program of this type is Napster, a utility for sharing audio files by manually registering them with a central directory service. Another popular program is Gnutella which shares more general-purpose files. The general term for both programs is a "peer-topeer file sharing service".

An additional application which has been developed based on this model is a distributed search engine. Operators of host computer sites wishing to permit searches register with the central directory service and then answer queries directed to them by that service. When a user performs a search, the central service receives the request, compares the request to information about the content of each host, and then transmits a copy of that request to all hosts which are able to satisfy the type of the request. The search results subsequently received from these hosts are then processed and sent to the requesting user. This is very similar to the functioning of existing search engines except that the searches are distributed to and performed by the individual hosts registered to a directory service rather than by the central site. This approach is commonly called a meta search engine.

SUMMARY OF THE INVENTION

Expanding on the above concepts, the invented system is a service which performs centralized searches based on index information transmitted by peer systems to the central site using an agent program running on each peer, and then directs the peer systems to each other for the purpose of retrieving files.

If none of the peer systems known to contain the file is online (and the file is therefore not available), the request is placed in a queue of file requests maintained by the central site, When a system containing the requested file connects to the service, the requested file is retrieved from that system and then distributed to the other systems which had requested the file. Files retrieved for systems not currently online are held in a queue until the user connects or are emailed to the user, usually as an email attachment. Or, when a computer system containing the file connects to the central site, the file is sent by the system containing the file either to the central site or directly to the user who requested the file via email attachment.

The indexing and content reporting functions necessary for the service are performed by an individual copy of an agent program downloaded and installed by each peer system user. This agent program is described in detail in pending U.S. patent application Ser. Nos. 09/419,405, U.S. Pat. No. 6,516,337, and 09/575,971, filed May 23, 2000, by the same inventors which are hereby incorporated by reference. The indexing process on each system may be initiated manually or on a scheduled basis, with updates transmitted whenever the user connects to the central service.

The agent is also responsible for transmitting copies of the requested file to the systems whose requests are waiting in the queue and picking up copies of files from the queue it had previously requested.

Unlike competing prior art systems, this agent-enabled system is able to maintain a central searchable index of the contents of the files, which is always available to users whether or not the site reporting the information found in the index is on-line.

This invention has great application not only in the general Internet market, but also in intranet markets where many users maintain local copies of files. It is also extremely useful for communities of users who wish to exchange similar information, or for mobile users who are not always able to be online at opportune times. This invention allows users to share files without having a web page.

This invention also allows the identity of each contributor of a copy of a file to remain anonymous. Only the central server knows the internet address and other identifying information about each contributor, and this information is stripped from each file before the file is forwarded.

This system also allows the sharing of files by systems which are protected by a secure firewall. The firewall prevents computers on the inside from serving files in response to conventional requests from the outside, but it allows the sending of an email with an attachment. To allow operation of the invented file sharing system without compromising the firewall, the agent program is configured to behave as follows. The agent reports to the central server the identities of files on the computer that will be provided if requested by others. When an email request for a file is received by the agent from the central server, the agent generates an email in response, attaching the requested file if that file is still on a list of files that may be provided by the agent.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a functional block diagram of a conventional search engine for the world wide web.

FIG. 2 is block diagram showing the architecture of a search engine for actively indexing the world wide web according to one embodiment of the present invention.

FIG. 3 is functional block diagram of the central server of FIG. 2.

FIG. 4 is a bubble chart illustrating the generation and processing of a brochure file in the indexing system of FIG. 2.

FIG. 5 is a bubble chart illustrating the process of the agent program in updating itself along with the local index generated by the agent program.

FIG. 6 is a bubble chart illustrating the process executed by the queue manager of FIG. 3 in queuing update entries and transferring these entries to the remote queue manager of FIG. 3.

FIG. 7 is a bubble chart illustrating the process executed by the update process server of FIG. 3.

FIG. 8 is a bubble chart illustrating the overall data flow in the search engine of FIG. 3.

FIG. 9 is a functional block diagram of a distributed search engine according to another embodiment of the present invention.

FIGS. 10 and 11 are diagrams illustrating operation of a distributed accounting and inventory system on an intranet according to one embodiment of the present invention.

FIGS. 12–44 are figures illustrating components of the indexing system of FIG. 2 for a Java-based implementation of the indexing system according to one embodiment of the present invention.

FIG. 45 is a functional data flow diagram illustrating an alternative embodiment of the central cataloging site of FIG. 2.

DETAILED DESCRIPTION

This invention is preferably implemented as described in detail in pending U.S. patent application Ser. Nos. 09/419,405, U.S. Pat. No. 6,516,337 and 09/575,971, filed May 23, 2000, by the same inventors which are incorporated by reference.

A domain name service (DNS) maps names (domain names) to addresses (Internet Protocol(IP) addresses). Domain names are scarce and expensive to obtain and maintain. A secondary DNS system could be built for the peer-to-peer network using peer-to-peer agents and the central index. Content providers could choose names (agent names) and those name would be associated with an agent indexing their site. Then, these names could be made known to others without providing the IP addresses, and the IP address can change and the content could still be found provided the agent name is not changed.

FIG. 2 is a block diagram of an indexing system 200 for actively indexing the Internet according to one embodiment of the present invention. The system 200 includes a central server 202 that stores a central index and processes search queries received over the Internet and also includes agent programs or agents 204 that reside on respective remote servers 208 and operate to provide periodic index updates to the central server 202, as will be described in more detail below. The system 200 also includes brochure files or brochures 206 residing on respective remote servers 208, each brochure file containing non-HTML or conceptual information about the Web site for use in generating the central index on the server 202, as will also be explained in more detail below. For the sake of brevity, only two remote servers 208 and the corresponding agents 204 and brochures 206 are shown in FIG. 2. The system 200, however, includes numerous such remote servers 208, agents 204, and brochures 208, as will be understood by those skilled in the art.

Each of the components in the central server 202 will now be described generally, with these respective components being described individually in more detail below. The central server 202 includes a router 210 that directs packets comprising search requests and update transactions through a load balancing switch 212 to an appropriate set of servers 214, 230 and 222. The switch 212 balances traffic to all web servers 214 to prevent overloading respective web servers and improve overall performance of the central server 202. The router 210 also functions to allow offline updates of index server sets 216 and as a dispatch point to prevent searches from being applied to an index server currently receiving updates, as will be explained in more detail below. The web servers 214 receive and preprocess index queries and receive and process brochure 206 generation or modification requests. In addition, the web servers 214 generate the parallel queries necessary to perform a search using the index servers 216. In one embodiment of the central server 202, there are twenty web servers 214.

The central server 202 further includes a master index server 218 containing a master copy of the entire central search index or catalog. In the embodiment of FIG. 2, the master index server 218 has a redundant array of independent disks or RAID 5 to provide protection against disk failures and loss of the central search index. In addition, the central index stored on the master index server 218 is also stored on a remote master index server 220 at a different physical location to provide backup of the central search index.

A number of update servers 222 each receive updates from the agent programs and store the current version of the agent program for download and update of the local agent programs, as will be described in more detail below. In addition, the update servers store the digital signature of the agent program and also store the remote web hosts' last local index, which are utilized during the updating of the remote agent program and during updating the local index, as will also be discussed in more detail below. Each of the update servers 222 applies all index change transactions through a firewall/router 224 to the master index server 218 which, in turn, updates the central search index and then distributes those changes to the various index servers sets 216. The master index server 218 also sends instructions to the Name Space/Directory Server 233 to dynamically determine which set of index servers 216 is to remain on-line to service search requests, and which set is to receive the updates.

The central search engine 202 further includes a brochure database server 226 and brochure check server 228. The brochure database server 226 stores a brochure database as a list of brochures and their associated data fields for each web site. The web servers 214 may request records from or add records to this brochure database depending on the actions taken by web site administrators while maintaining their brochure entries. The brochure check server 228 periodically checks for valid new brochures as defined within the brochure database server for web sites that are not being processed by a local agent program, as will be described in more detail below. If the defined brochure in the brochure database server 226 is not found by the brochure check server 228, a notification is sent to the administrator of the site where the brochure was supposed to be found.

When a brochure file is requested for a site which is not served by an agent 204, a message is sent to the Internet Service Provider ("ISP") or system administrator for the site hosting the web site, indicating that users of the system are requesting brochures. This server also periodically checks the validity of existing brochures on all sites and notifies the web site administrator if a brochure file is missing. If a brochure is missing and remains missing for a given number of check cycles, the brochure check server 228 sends a request to the brochure database server 226 to delete the entry for the brochure. The brochure check server 228 detects any changes in brochures, such as additions or removals, and converts these changes to transaction batches that are forwarded to a queue manager which, in turn, applies these changes to update the central index on the master index server 218, as will be described in more detail below.

The brochure check server 328 periodically verifies the status of all brochures at sites that are not being indexed by an agent 204.

The components of the central server 202 and their general operation have been described, and now the opera-

tion of the agent **204** and brochure **206** will be described in more detail. The agent **204** and brochure **206** may both be present at a remote server **208**. A brochure **206** and agent can function independently of each other, as will be discussed in more detail below. The agent **204** is a small local program which executes at the remote server **208** and generates an incremental search engine update for all of the participating web sites on the web host **208**. These index updates are transmitted by the agent **204** to the central server **202**, where they are queued for addition to the central index.

The agent **204** runs on a system, such as a web host server, at the site of an organization, and processes content (objects) for all web sites available via mass storage from that system. The agent **204** processes all web sites located within the mass storage area to which it has access, unless configured to exclude some portion of a site or sites. The agent **204** uses the local web server configuration (object catalog or file system information) data to determine the root directory path (or other location information for the particular file system) for all web site file structures available. The agent **204** reads files directly from local mass storage, and indexes the keywords from the files and meta data about the files. In contrast, a spider program, as previously discussed, is located on a server remote from the local site and renders each web page file before tokenizing and parsing each page for indexing. The agent **204** follows the structure of the local mass storage directory tree in indexing the files, and does not follow uniform resource locators ("URLs") stored within the HTML files forming the web pages. Since the agent **204** is present at the remote server **208** and has access to files stored on the server's mass storage, the agent is potentially capable of retrieving non-html data for indexing from these locally stored files, such as database files and other non web-page source material. For example, a product catalog stored in a database file on the remote mass storage may be accessed and indexed by the agent **204**.

While indexing the web sites at the remote server **208**, the agent **204** recognizes brochures **206** stored at web sites on the server and provides index updates based on the contents of the brochures found. Once the agent **204** has indexed the web sites at the remote server **208**, the agent transmits a transaction list to the central server **202**, and this transaction list is stored on one of the update servers **222**. The transaction list is referred to as a batch, and each batch contains a series of deletion and addition transactions formatted as commands. More specifically, each batch represents an incremental change record for the sites at the remote server **208** serviced by the agent **204**. The update server **222** thereafter transfers each batch to the master index server **218** which, in turn, updates the master index to reflect the index changes in the batch. It should be noted that the agent **204** transmits only "incremental" changes to the central server **202**. Conversely, a conventional spider program requests the entire rendered HTML page from the remote web site via the remote server **208**, and then parses the received page for keyword information.

The brochure **206** is a small file that may contain conceptual and other non-HTML information which would be useful to improve the indexing of sites or parts of a single site on the remote server **208**. A brochure **206** may contain any information pertinent to the web site, including but not limited to keywords, phrases, categorizations of content, purpose of the site, and other information not generally stored in a web page. The brochure **206** is generated manually by individual web site administrators. The administrator fills out a form at the central server **202**, and receives an email containing the brochure **206** or downloads the brochure after submitting the form contents. Upon receiving the brochure **206**, the administrator stores it within the file structure of the web site on the remote server **208**. There may be multiple brochures **206** at the same web site, each describing specific portions of the site. Each brochure **206** may refer to a single web page or a group of web pages stored within a specific subdirectory at the web site. All information stored in each brochure **206** is applied to the pages referenced in the brochure.

The overall operation of the central server **202** will now be described in more detail with reference to the functional block diagram of FIG. **3**. In FIG. **3**, many components previously discussed with reference to FIG. **2** are shown, and for the sake of brevity the detailed operation of each such component will not again be described in detail.

In operation, the central server **202** performs three primary functions: 1) processing search queries from remote users; 2) brochure generation and verification; and 3) index update processing. In processing search queries from remote users, the Web servers **214** receive search queries from remote user browsers. A router, which corresponds to the routers **210** and **212** in FIG. **2**, directs the search query to the appropriate web server **214**. The web server send the query to a Query Processor **234** which parses the query and sends it to the available index server set **216** or **217** as listed in the Name Space Server **233** for appropriate segment of the index. The selected index server sets **216** or **217** thereafter return search results to the query processor in response to the applied search query, and these search results are sent to the Web server **214**, which, in turn, returns the search results to the remote user browser.

The central server **202** also allows remote users to generate and download brochures **206** to their remote site, and also verifies the validity of brochures **206** on Web sites not serviced by an agent **204**, as will now be explained in more detail. The Web servers **214** receive and process brochure **204** generation or modification requests from user browsers. Once the brochure **204** has been generated or modified, the brochure is transferred to the brochure database server **226**, which stores all existing brochures. The brochure check server **228** periodically checks for new brochures **206** stored on the brochure database server **226** for Web sites that are not served by an agent **204**. When a brochure **206** is requested for a Web site which is not served by an agent **204**, the brochure check server **228** sends a message to the system administrator or Internet service provider for the server hosting a Web site telling them that site administrators on their server are requesting brochures **206**. The brochure check server **228** also periodically verifies the validity of existing brochures **206** on all sites not serviced by an agent **204**. If a brochure **206** is missing for a predetermined number of verification cycles, the brochure check server **228** instructs the brochure database server **226** to delete the entry for that brochure. The brochure check server **228** also converts any modifications, additions, or deletions to brochures **206** to transaction batches, and forwards these transaction batches to a queue manager **302**. The queue manager **302** receives brochure update transaction batches from the brochure check server **228** and also receives agent update transaction batches from the agent update server **222**, as will be described in more detail below.

The central server **202** also performs index update processing to update the central index stored on the master storage server **218** and the segmented central index stored on the index servers **216**, **217**, as will now be described in more detail. As described above, the queue manager receives update transaction batches from the brochure check server

228 and the agent update server 222. The agent update server 222 receives queries from the agent as to the current state of the agent's version and the status of the last index updates of the site. If the agent is not of a current version, a current version is automatically transmitted and installed. If the state of the site indexing is not consistent as indicated by a match of the digital signatures representing state of the site and the state of the central index the last time an update was received and successfully processed and added to the central index, then the agent will roll back to previous state and create the necessary additions and deletions to the state of the site and the central index into agreement. The agent 204 will then sent the additions and deletions along with a current digital signature to the queue manager 302 The queue manager 302 receives incremental index updates from the agents 204 present on the remote servers 208 and converts these updates into update transaction batches which, in turn, are transferred to the update processing server 306. The queue manager 302 stores the received update transaction batches, and periodically transmits a copy of the stored transaction batches to a remote queue manager 304 for processing by update processing server 306 and being applied to the remote master storage server 220. The queue manager 302 also periodically transmits a copy of the stored transaction batches to and update processing server 306. The queue manager 302 stores update transaction batches received from the agent 204 during a predetermined interval, and upon expiration of this interval the update batches are transferred to the update processing server 306. Upon receiving the update transaction batches the update processing server 306, applies all the batches to update the central index stored on the master storage server 218. Once the central index stored on the master storage server 218 has been updated, the master storage server 218 applies the update transaction batches through the router to update the segmented central index stored on the index server sets 216, 217.

During updating of the segmented central index stored on the index server sets 216, 217, the update transaction batches are directed to only one set of index servers 216, 217 while the other set remains online to handle search queries, and thereafter places the updated set of index servers 216, 217 online and updates the set previously online. For example, assume the index servers 216 are the primary set of index servers and the servers 217 are the secondary set. Each index server set 216, 217 can contain all or a portion of the central index 218. As seen from the above example, the primary and secondary index server sets 216 and 217 eliminate the need for record locking of the segmented central index to which search queries are applied. Thus, all records of the segmented central index are always available for search queries. Moreover, if one server of the primary index server set 216 or 217 fails, the remaining servers of that set will continue to serve queries. If the entire server set fails, the corresponding secondary index server set is made the primary so that the entire segmented central index is available for applied search queries. It should be noted that in the unlikely event that both the primary and secondary index server sets 216, 217 for a particular segment of the central index simultaneously fail, the remaining segments of the central index, remain available for applied search queries, and only the segment of the central index stored on the failed index servers becomes unavailable. In other words, search queries are still applied to the vast majority of the central index so that reasonable search results may are still obtained. In a case were both server sets fail, queries for the segment that had failed could be sent to central index.

The index server set or sets are used to provide query results for searches submitted by the Web Servers. Each set of servers is identical, and each set of servers contains a portion of the overall index. Initially, the division will be alphabetical and numerical, for a set of 36 servers. Server "A" would contain the index for all words beginning with "A". Only one set of servers is updated at a given time, while the other set remains on-line to service search requests. This permits the system to be run without file-locking constraints and allows for fail over should a server become inoperative.

FIG. 4 is a bubble chart illustrating the generation and processing of a brochure 206 in the indexing system 200 of FIG. 2. As previously mentioned, the purpose of the brochure 206 is to allow the web host 208 and the web site to provide specific non-HTML information, which will help the central server 202 in indexing the site and in order to provide more relevance to query results. The brochure 206 can be created in two ways. First, as part of the installation program for the agent 204, the administrator of the remote server 208 completes a form that is converted to an encoded brochure file 206, and then copied into the web directory on the remote server 208. This method of generating the brochure 206 will be discussed in more detail below. The second method of generating the brochure 206 utilizes a brochure creator interface on the web servers 214 at the central server 202. This method will now be described in more detail with reference to FIG. 4.

To create a brochure 206 using the brochure creator interface, a user's browser 400 applies a brochure generation request 402 to the associated central site web server 214. In response to the request 404, the brochure creator interface generates a form which the user completes, and then sends a brochure request 406 to the brochure server 226, which generates an encoded brochure file that is then sent to the central site web server 214. The central site web server 214 then sends the encoded brochure file to the user's browser 400. The encoded brochure file 206 is then stored in local storage 408. Subsequent to receiving the encoded brochure file 206, the user sends the encoded brochure file 206 via the user's web browser 400 to the web host site storage 410 (e.g., the web site host computer).

The brochure server 226 stores the brochure data 407 in a brochure database 424 on the central server 202 once it has been generated as a result of a brochure generation request 404. To verify proper storage of encoded brochure files 206, the brochure check server 425 retrieves brochure data 420 from the brochure database 424 and sends a request 416 to the web host server 404 to retrieve the encoded brochure file 206 from the web host site storage 410. Upon successful retrieval of the brochure file 206, the brochure check server generates and transmits object references 422 created as a function of the brochure data 420 to the queue manager 302. The queue manager 302 thereafter updates the central index to include the generated object references.

The directory structure of the host and web site are used to determine the relevance of the information in the brochure. Information in a brochure located the root directory will apply to all sub-directories unless superceded by another brochure. Information in a directory brochure will apply to all subdirectories unless superceded by information in a subdirectory brochure. Where a brochure is placed determines for which content the information applies. A web site owner can have as many brochures as there are pages or directories in his site. A site owner can request that their site be excluded from the Index by checking the EXCLUDE box next to the URL and copying the brochures into the directory to be excluded.

The host uses the configuration section of the agent program to create site brochures, and can create site brochures for an entire IP address or for any subsection of the site.

In addition to the host brochure, a web site owner may also place a site brochure on his web site. The purpose of the site brochure is to allow the web site owner to provide specific conceptual or non-html information, which will help in indexing their site.

The web site owner can create a different site brochure for each page or directory on the site. For example, if the web site includes pages in different languages, the web site owner should create a site brochure for each language with keywords and categories that match the language. Once the web site owner has filled in the brochure form, they will click a button on a web page from the web server at the central server, and a web server creates an encoded html file that is then sent or downloaded to the site owners computer. Each encoded brochure file could be given a particular name, such as brochure-domainname-com-directory-directory-directory.html, and the site owner is instructed to copy the encoded file into the specified web directory on the site.

At anytime, the web site owner can visit the central server site, update their brochure, and download a new encoded brochure. When updating an existing brochure, the current brochure information for the URL entered will be displayed to reduce input time. Any site brochure will supercede the host brochure information, and information contained in the site brochure will be assumed to be more current and accurate and will be used by the agent for indexing purposes. A site brochure that is farther down in the directory tree from the root directory will supercede a site brochure that is above it in the directory tree. A site owner can request that their web site be excluded from the index by checking the EXCLUDE box next to the URL and copying the brochures into the directory to be excluded.

If the host or web site URL is not currently being indexed, the web server performs the following operations. First, an automatic email is sent to contacts at the host to encourage the host to install the agent. An automatic email is also sent to a contact person for the web site with a "Thank You" and a request that they ask their host to install the agent. In addition, a retrieval order is generated for the central server to retrieve the brochure file from the web site in one hour. If the retrieval order is unsuccessful, it will be repeated 2, 4, 8, 24 and 48 hours later, until successful. If still unsuccessful after 48 hours, the retrieval order is canceled. By verifying the presence of the site brochure in the specified location, unauthorized information about a site may not be created by a third party in an attempt to have their site indexed along with a more popular site. This is a common problem with existing search engines where a third party copies the keywords from a meta tag in a popular site. The bogus site with copied keywords is then submitted to a search engine for indexing, and when search queries are applied to the search engine that produce the popular site the bogus site is also produced. This may not be done with the site brochure because the brochure is not an html page available to outside persons and because it is encrypted so even if the file is obtained the information contained therein is not accessible.

Software to create brochures and agent programs will be distributed free to software publishers for inclusion in their web authoring software and to web server manufactures, publishers and OEMs for pre-loading on or inclusion with their products.

FIG. **5** is a bubble chart of the process executed by the agent **204** according to one embodiment of the present invention. As previously mentioned, the agent **204** periodically executes the illustrated process to update itself and to update the corresponding local index, as will now be described in more detail. The process begins in step **500** in which the agent verifies that it is the most current version of the agent program. More specifically, in step **500** the agent sends a request **502** to one of the update servers **222** for the digital signature of the current version of the agent program. The update servers **222** returns the digital signature **504** for the most current version of the agent. In step **500**, the digital signature hash of the local agent is compared to the returned digital signature hash to determine Whether the local agent is the most current version. In other words, if the two digital signatures are equal, the local agent is the most recent version, while if the two are not equal the local agent is an outdated version of the agent program and must be updated. When the two digital signatures are unequal, the program goes to step **506** in which the most current version of the agent program **508** is received from the update server **222**. Once the local agent program has been updated, the program proceeds to step **510**. Note that if the digital signature of a local agent program is equal to the digital signature **504** of the most recent version of the agent, the program proceeds directly from step **500** to step **510**.

In step **510**, the agent program compares the digital signature hash for the existing local index previously generated by the agent program to the digital signature hash stored on the central server **202** for the existing local index. The agent program performs this step to synchronize the local index and the remote local index stored on the central server **202** by ensuring the digital signature of the existing version of the local index matches the digital signature for the existing version of the remote local index. If the two digital signatures are equal, the agent program goes to step **512** and generates and updated local index by evaluating, such as by tokenizing and parsing, local files **513** on the web host serviced by the agent. Once the updated local index has been generated, the agent program proceeds to step **514** where the updates along with the digital signature of the new local index are transferred to the agent update server **222** on the central server **202**.

If step **510** determines the two digital signatures are not equal, the agent program goes to step **516** to roll back to a previous state that matches the local files **513** or to generate a completely new local index for the web host serviced by the agent. After the complete new local index is generated, the agent program once again proceeds to step **514** and the updates are transferred to the agent queue manager **302**. As previously mentioned, comparing the digital signatures in step **510** synchronizes the local index and remote local index. Furthermore, this step enables the agent program to rebuild a completely new local index for the site serviced by the agent program in the event the index is lost at the central server **202**. Thus, should the central server **202** crash such that the central index is corrupted and non-recoverable, the agent programs at the each remote web host will rebuild their respective local indices, and each of these local indices will be transferred to central server **202** so that the entire central index may be reconstructed.

As mentioned above, the agent **204** is a software program that a web host downloads from the web servers **214** and installs on the host's server. To install the agent **204**, the host runs an agent installation program, which collects information about the web site host and about the site itself, and also creates the web site host's brochure **206** of non-HTML information. As part of the installation, the site host schedules a preferred time of day for the agent **204** to automati-

cally index the web site and transfer index updates to the central server **202**. The agent and the queue manager can work independently or together to reschedule when to perform and transmit the site update. Resource availability is the primary and any other factor, which may effect the quality or efficiency of the operation may be used by the agent and the queue manager in rescheduling updates.

In the preferred embodiment the agent **204** initiates all communications with the central server over a secure socket authorized and setup by the site host. But the central server **202** could also initiate communications or trigger actions of the agent or retrieve data process by the agent. All data and program updates sent between the site host and the central server are sent in compressed and encrypted form. During the normal index updating process, the agent **204** is automatically updated, as will be explained in more detail below. The site host may receive a daily email saying the site had been properly updated or that no update was received and no action is required. The agent **204** also maintains a log of indexing activity and errors encountered, and this activity log can be viewed by the site host by opening the agent **204** and accessing the log. Although the agent **204** automatically indexes the sites on the host at scheduled times, the host can at anytime initiate an indexing, update by opening the agent **204** and manually initiating an index update.

In operation, the agent **204** verifies that the agent program is current and that the site index matches the last update received and successfully added to the central index on the central server **202**. After verification and updating of the agent **204** if required, the agent checks the site for new, modified or deleted files. The new or modified files are indexed and the information added to or deleted from the site index or a list of additions and deletions transactions are created. The incremental changes to the site index along with a digital signature of the entire site index are sent to the central server **202** and the results logged in a site activity log maintained by the agent **204**. The agent **204** is run by either being manually started by the site host or automatically started by a scheduler component of the agent.

It is not necessary that a local index be maintained at the site but only that a list of digital signatures representing the site at the time of the last update be maintained. The digital signature could be used to determine whether the local site and the central index are properly synchronized and which portion of the site had changed since the last successful update. Then instructions to delete all references from the central index **218** to files located at the web host that have changed or which no longer exist would be sent by the agent to the queue manager. New references would then be created for all new or modified files and would be sent by the agent to the queue manager as additions to the central index **218**.

The process executed by the agent **204** will now be described in more detail. The agent **204** first checks with the central server **202** for the current version of the agent program. More specifically, the agent **204** calculates a digital signature of the agent program files and contacts the central server **202** over a secure socket. The agent **204** then requests a digital signature of the current version of the agent program files located at the central server **202**, and compares the two digital signatures. If the two signatures match, the version of the agent **204** is current and no update is required. When the two signatures do not match, the current version of the agent **204** is downloaded from the central server **202**. Once the current agent **204** is successfully downloaded, the new agent program files are installed and the agent restarted.

At this point, the agent **204** begins the process of updating the index of the local site. First, the agent **204** determines whether the last index update was completed and transmitted successfully. If not, the agent **204** renames the Old-Site-Index file to Site-Index and the Old-Site-File-List to Site-File-List. The agent **204** then calculates a digital signature for the Site-Index file and a signature for the Site-File-List file and compares each to the digital signatures created at the end of the last successful update for Site-Index and Site-File-List files. If the digital signatures match, the agent **204** sends them to the central server **202** for comparison and waits for confirmation.

If the central server **202** does not confirm the match of the digital signatures (i.e., the signatures for the Site-index and Site-File-List files on the central server **202** do not match those on the remote site), the agent **204** deletes the Site-index and Site-File-List files, and notifies the central server **202** to delete all site records. Next, if the agent **204** was updated and Fields were added or deleted from the Site Index file, then the agent updates the Site-Index file to include the updates. The agent **204** then determines if the Site-File-Lists file exists, and renames the Site-File-List file to Old-File-List and create a text file named Site-File-List. If no Site-File-List exists but Old-File list exists, the agent **204** copies the Old-File-List file to Site-File List. If no Site-File-List and no Old-File-List files exist, the agent **204** creates a text file named Site-File-List. The agent **204** then calculates a digital signature hash for each file on the site and the host brochure and records the file name including full path and digital signature hash of all files.

If the central server **202** verifies that the digital signature hash of the Site-Index file and the digital signature hash for the Site-File-List file match, the agent **204** verifies the brochure files. More specifically, the agent **204** determines if the file brochure.html file name does not match the directory in which it is located. If the file brochure.html is not in the expected directory, the agent **204** sends a warning email to the site contact listed in the brochure, and then renames brochure.html to WrongDirectorybrochure.html.

If the agent **204** determines that all brochure.html files match the directory in which they are located, the agent **204** deletes a file named Exclude-File-List, creates a text file named Exclude-File-List, checks brochures for EXCLUDE sites flags, and adds file names of files to be excluded from the index to the Exclude-File-List file. The agent **204** then creates a Deleted-File-List file containing a list of files that no longer exist on the site in their original location. More specifically the agent **204** deletes the old Deleted-File-List file, creates a text file called Deleted-File-List, compares the Site-File-List file to Old-File-List file and records in the Deleted-File-List any files in the Old-File-List that are not in Site-File-List.

The agent **204** then creates a New-File-List file containing a list of files that where created or modified since the last update. To create the New-File-List file, the agent **204** deletes the current New-File-List file, creates a new text file called New-File-List, compares the file Site-File-List to the file Old-File-List and the file Exclude-File-List, and records in the New-File-List file any files in Site-File-List that are not in the Old-Site-File-List or in Exclude-File-List files.

Next, the agent **204** indexes the corresponding site and creates a new Site-index file. More specifically, the agent **204** determines if the Site-Index file exists, and, if yes, copies the Site-Index file to an Old-Index file. If the Site-Index file does not exist, the agent determines if the file Old-Site-Index exists, and if yes copies the Old-Site-Index file to Site-Index file. If Old-Site-Index file does not exist, the agent **204** copies a Sample-Site-Index file to the Site-Index file.

The agent **204** then creates a New-Records-Index file and a Deleted-Records-List file. The agent **204** next removes records of deleted or modified files from the Site index. More specifically, the agent **204** deletes all records from Site-Index for files in New-File-List, deletes all records from Site Index for files in Deleted-File-List, and records the Host IP, URL, and record ID Numbers for each record deleted into Deleted-Records-List.

The agent **204** then runs an indexing program against all files in the New-File-List file and creates a record for each new key word, phrase, MP3, Video, Movie, Link and brochure information and adds these to the Site-Index file. The agent **204** then copies each new record created to the New-Records-Index file. If new fields were added to the Site Index, the agent **204** runs the indexing program against all files for new field information and creates records in Field-Update-Index for all information found. The agent **204** then updates the Site-Index file from the Field-Update-Index file.

At this point, the Site-index file has been updated, and the agent **204** calculates a digital signature for the Site-index file. More specifically, the agent determines if the Update-Status file exists, and if so opens this file. If the Update-Status file does not exist, the agent **204** creates a text file called Update-Status and opens this file. The agent **204** then calculates the digital signature of the Site Index file, and records the Site-Index digital signature along with the date and time in the Update-Status file. Next, the agent **204** calculates the digital signature of the Site-File-List file, and records the Site-File-List digital signature along with the date and time in Update-Status file.

Finally, the agent **204** creates a Site-Map file for the sites serviced by the agent. More specifically, the agent **204** determines whether the Deleted-File-List or New-File-List contain files, and, if yes, the agent deletes the Site-Map file. The agent **204** then generates a site map for the Site-Map file from the Site-File-List. Once the Site-Map file has been generated, the agent **204** sends New-Records-Index and Deleted-Records-List files to the central server **202**. More specifically, the agent **204** opens a secure connection and contacts the central server **202**. The agent **204** then compresses the files to be sent, encrypts these files, and sends the compressed and encrypted files in the New-Records-Index, Field-Update-Index, Deleted-Records-List, digital signature for the Site-index, Site-Map, and the Site-File-List to the central server **202**, which the uses these files to update the central index. Once the agent **204** has successfully sent this information to the client server **202**, the agent **204** records the digital signature of the Site-Index file, the time of the successful transfer, the date and size of the files transferred in the Update-Status file, and thereafter deletes the sent files. The agent **204** then closes the secure connection to terminate the update process.

FIG. **6** is a bubble chart illustrating the process executed by the queue manager **302** of FIG. **3** in queuing update entries and transferring these entries to the remote queue manager **304**. The queue manager **302** receives update entries **600** from the agent update server **222** and update entries **602** from the brochure server **228**, and places these update entries in an update queue **604**. The entries in the queue **604** are transferred to a queue database **606**. Once the queue **604** is done receiving update entries **600, 602**, which may be when the queue is full or at predetermined intervals, the queue manager **302** goes to step **608** and retrieves the queue entries from the queue database **606** and sends them to the remote queue manager **304**. As previously described, the update entries stored in the queue database **606** are thereafter processed by the update processing server **306** (see FIG. **3**) to update the local master index on master index sever **218** (see FIG. **3**). The queue manager **302** also receives a deletion request (not shown) from the update processing server **306** and deletes update entries stored in queue database **606** in response to this deletion request, as will be explained in more detail below with reference to FIG. **7**. The queue functions are preferable implemented using a customized version of the standard UNIX email handlers, where each inbound email represents a request for a file or for the content of a file.

FIG. **7** is a bubble chart showing the process executed by the update processing server **306**. The process begins in step **700** with the update processing server **306** retrieving queue entries **700** from the queue manager **304**. In the embodiment of FIG. **7**, the queue entries **702** are retrieved periodically so that in step **700** the queue entries for the last N hours are retrieved. From step **700**, the process proceeds to step **704** and the update processing server **306** applies the queue entries to the master index server **218** which, in turn, utilizes the queue entries in updating the master index, as previously described. Once the queue entries **702** have been applied to the server **218**, the process proceeds to step **706** and the update processing server **306** applies a deletion request **708** to the queue manager **302** (see FIGS. **3** and **6**). In response the deletion request **708**, the queue manager **302** deletes the update entries stored in the queue database **606** that have now been applied to the master index server **218**. The central index on the master index server **218** has now been updated to include entries in the queue database **606**, so these entries are deleted since they are now reflected in the central index and thus no longer needed.

FIG. **8** is a bubble chart illustrating the overall data flow between the search engine **202**, agent, and brochure components of the active indexing system **200**. Each aspect of the overall data flow has already been described in a corresponding section above, and thus FIG. **8** will now be described merely to provide a brief description of the overall data flow of the indexing system **200** according to one embodiment of the present invention. The components of the process in FIG. **8** may logically broken into two functional groups, an indexing group and a searching group. In the searching group, a user **800** applies a search request to one of the web servers **214**, which processes the search request and applies it to selected ones of the index servers **216, 217**. In response to the applied search request, each of the search index servers **216, 217** queries its corresponding local index segment **802** and generates search data. The index servers **216, 217** then return the search results to the web server **214**, which, in turn, provides the user **800** with the search results corresponding to his applied search request.

The web servers **214** also handle version queries from agents **204** on source sites. Each agent **204** sends a version check **804** that is processed by one of the web servers **214**. In response to the version check **804**, the web server **214** returns the digital signature of the most recent version of the agent **204**, and also supplies the updated version **806** of the agent **204** to the source site if an update is required.

The remaining components in the FIG. **8** are in the indexing group. The queue manager **302** receives updates from each of the agents **204** and from the brochure check server **228**, which services sites without an agent **204** as previously described. The queue manager makes update and deletions to the queue database **602** corresponding to the received updates, and also provides a mirror copy of these updates to the remote queue manager **304**. The update processing server **306** retrieves the update entries from the queue manager **302**, and applies the updates to the master

index servers **218**. The server **218** updates the master index to include the applied updates, and the update processing server **306** then sends a deletion request to the queue manager **302** to delete the corresponding entries from the queue database **602**.

Once the master index server **218** has updated the master index, the server updates the segmented index stored on the search index servers **216, 217** as previously described. Each of the search index servers **216, 217** updates its corresponding portion of the segmented index in response to the updates from the master index server **218**. As previously mentioned, the entire segmented index stored on the index servers **216** is continuously available for processing search requests even during updating of the segmented index. The entire segmented index is available due to the redundant architecture of the servers **216, 217**, as previously described.

FIG. **9** is a functional block diagram of a distributed search engine **900** according to another embodiment of the present invention. The search engine **900** includes a central search engine **902** connected over a network **904**, such as the internet, to a plurality of agents **906**, each agent being resident on a respective server **908**. Each agent **906** generates a list of digital signatures related to retrievable information on the corresponding server **908** and provides these signature to the search engine **902** which determines which files to access for updating its index, as will now be explained in more detail. In the following description, the server **908** is a standard web server, but one skilled in the art will appreciate that the distributed search engine **900** can be implemented for a number of other services available on the internet, including but not limited to email servers, ftp servers, "archie", "gopher" and "wais" servers. Furthermore,, although the agent **906** is shown and will be described as being on the web server **908**, the agent **906** need not be part of the program which processes requests for the given service.

In operation, the agent **906** periodically generates a list of signatures and accessible web pages, which are then stored on the local web server **908**. The digital signature generated by the agent **906** could be, for example, an digital signature of each file on the server **908**. The list of digital signatures is then transmitted by the agent **906** to the search engine **902**, or the search engine **902** may retrieve the list from the servers **908**. A digital signature processing component **910** in the search engine **902** then compares the retrieved digital signatures against a historic list of digital signatures for files on the server **908** to determine which files have changed. Once the component **910** has determined which files have changed, a spider **912** retrieves only these for indexing.

The digital signatures may be stored in an easily accessible file format like SGML. Alternatively, the digital signatures could be generated dynamically when requested on a page by a page or group basis. This would insure that the signature matches the current state of the file. In addition, several new commands would be added to the standard http protocol. The new commands perform specified functions and have been given sample acronyms for the purposes of the following description. First a command GETHSH retrieves the digital signatures for a given URL and sends the signatures to the search engine **902**. A command CHKHSH checks the retrieved digital signature for a given URL against a prior digital signature and returns TRUE if the digital signatures are the same, FALSE if not the same, or MISSING if the URL no longer exists. A command GETHLS retrieves a list of the valid URLs available and their associated digital signatures, and a command GETLSH retrieves the digital signature of the URL list.

Using the above command set, the search engine **902** need not request the entire contents of a page if that page has already been processed. Furthermore, there is no need to "spider" a site. Instead, the web server **908** provides the valid list of URLs which can then be directly retrieved. As an example, consider the following programmatical steps from the point of view of a search engine. First, given a web host **908**, fetch the digital signature of the URL list. If the digital signature does not match a prior digital signature for the list, fetch the list of URLs from the web server. Thereafter, compare the list of URLs at the client web server **908** just retrieved to those stored locally at the search engine **902**. From this comparison, a list of changed URLs is determined. The URLs that have changed are then retrieved and parsed for keyword and other indexing information. Once the indexing information is obtained, all URL's which do not appear in the retrieved list and the prior list are deleted from the search index on the search engine **902**.

From the above description, one skilled in the art will appreciate that it is not necessary to retrieve all pages on the web site for every indexing process. Full retrieval of all web pages is necessary only once or if the entire site changes. This has several effects, the most important being that the amount of information transmitted is drastically reduced. The above method is but one possible implementation or embodiment. In another embodiment, a list of URLs on the search engine could be used and the individual checking of web pages done using the commands given. For example, the search engine **902** could tell if a page was current by simply retrieving its signature. If current, no other activity is required. Otherwise, the page might be deleted if no longer present or re-indexed if it has changed.

All content from a single agent/site could be searched for by a peer system user using the agent name. The search results could then be displayed to the user in a dynamically created "home page" for the content provider identified by that agent name. The dynamic home page would include a listing of every item indexed by the agent with that agent name and the item titles would be displayed along with their descriptions.

In a conventional search engine, the search engine normally requests that a web server deliver HTML documents to the search engine, regardless of whether the contents of the page have changed since the last recursive search. This is wasteful not only of CPU resources, but very wasteful of bandwidth which is frequently the most valuable resource associated with a web site. Thus, current search engines and content directories require regular retrieval and parsing of internet-based documents such as web pages. Most search engines use a recursive retrieval technique to retrieve and index the web pages, indexing first the web page retrieved and then all or some of the pages referenced by that web page. At present, these methods are very inefficient because no attempt is made to determine if the information has changed since the last time the information was retrieved, and no map of the information storage is available. For example, a web server does not provide a list of the available URLs for a given web site or series of sites stored on the server. Secondly and most importantly, the web server does not provide a digital signature of the pages available which could be used to determine if the actual page contents have changed since the last retrieval.

Another alternative embodiment of the process just described is the automated distribution of a single web site across multiple servers. For example, a web site would be published to a single server. Periodically, a number of other servers would check the original server to see if any pages

have been added, removed or changed. If so, those pages would be fetched and stored on the requesting server. Another alternative embodiment is the construction of meta indexes generated as lists of URLs from many different web servers. Such a meta index would be useful as a means of providing central directory services for web servers or the ability to associate sets of descriptive information with sets of URLs. The method could also be used to create directory structure maps for web sites, as will be appreciated by one skilled in the art.

The indexing system **200** may be used not only on the global communications network but on corporate Intranets as well. A typical corporate intranet includes a central location, such as a corporate headquarters, at which a central searchable database is maintained, and a number of remote locations, such as regional offices or stores, coupled to the central location through a network of intranet. Each remote location transfers data to the central location for storage in the central database. The remote locations may also search the central database for desired information.

In transferring data from each remote location, data is typically stored at the remote location and then transferred to and replicated at the central location. One of four methods is generally used to update the central database, as previously discussed above under the Background section. First, all remotely stored data is copied over the intranet to the central location. Second, only those files or objects that have changed since the last transfer are copied to the central location. Third, a transaction log is kept at the remote location and transmitted to the central location, and the transaction log this then applied at the central location to update the central database. Finally, at each remote location a prior copy of the local data is compared to the current copy of the local data to generate a differential record indicating changes between the prior and current copies, and this differential record is then transferred to the central location and incorporated into the central database.

Each of these methods relies on duplicating the remote data, which can present difficulties. For example, redundant hardware at the remote and central locations must be purchased and maintained for the storage and transfer of the data over the intranet. Data concurrency problems may also arise should transmission of differential data from the remote locations to the central location be unsuccessful or improperly applied to the central database. Furthermore, if the intranet fails, all operations at remote locations may be forced to cease until communications are reestablished. A further difficulty is the author's loss of authority over his document and the responsibility for retention and data management decisions. In a centralized intranet, unregulated retrieval of objects from the central database to local storage can creates version control problems. Difficulty in handling revisions to an object may also arise in such a centralized system, with simultaneous revision attempts possibly causing data corruption or loss. Finally, in centralized system the size of the central database can grow to the point where management of the data becomes problematic.

With the architecture of the indexing system **200**, everything, including each field in a local database, is treated as an object. Instead of copying each object to a central location, an object reference is created at each local site and sent to a cataloging location or locations. The objects are not duplicated in a monolithic central database. One advantage to this architecture is that the decision of whether to expose the existence and classification of local objects becomes the responsibility and choice of the author, rather than a generic decision. In the system **200**, the implementation of retention rules and the physical location of the objects remain with the author. The searchable central catalog merely references the distributed objects, eliminating the need to make full copies and therefore manage a large storage system. Each local site generates and transfers information to the central server **202**, or to a plurality of central servers for use in a searchable catalog.

FIGS. **10** and **11** are diagrams illustrating operation of a distributed accounting and inventory system on an intranet **1000** according to one embodiment of the present invention. In FIG. **10**, the intranet **1000** includes three different physical locations **1002**, **1004**, and **1006** including catalogs **1008**, **1010**, and **1012**, respectively. Each location **1002–1006** also includes a source of objects (not shown in FIG. **10**) that corresponds to an inventory of items at that location. The sources objects or sources for the locations **1002**, **1004**, **1006** are designated sources **1002**, **1004**, and **1006**, respectively, in records of the respective catalogs **1008–1012**. In the example of FIG. **10**, the source **1006** is empty (i.e., no inventory items at location **1006**). Each of the catalogs **1008–1012** is a catalog of object references to objects in the source at the corresponding location and to objects at the other locations. For example, the catalog **1010** at location **1004** includes a record for part no. 1, which is part of the inventory or source **1004** at this location. The catalog **1010** further includes an object reference, as indicated by the arrow **1014**, for part no. 3, which is part of the inventory or source **1008** at location **1002**. The catalog **1010** does not store a duplicate copy of the information in the record for part no. 3, but instead merely stores a reference to that object.

FIG. **11** is another diagram of the intranet **1000** expressly illustrating the sources **1002–1006** on the locations **1002–1006**, respectively. The source **1006** is shown as containing no objects, such as may be the situation where the location **1006** is at a headquarters of a corporation.. The sources **1002** and **1004** each include objects or inventory items, such as where these locations are remote offices of the corporation. This example illustrates that records for objects are not duplicated on each location **1002–1006**, but instead object references in each of the catalogs **1008–1012** point to objects stored in remote sources.

The intranet **1000** provides several advantages in accounting or inventory control applications, and others. A conventional intranet requires the centralization of the catalog for purposes of control. The intranet **1000** separates the control of the physical inventory (objects in the sources **1002-1006**) from accounting control. Since the whole intranet includes only objects and object references, then central reporting and planning can occur to the location **1006**, but such reporting merely corresponds to data being read from the remote locations **1002**,**1004**, and no data is modified. In the intranet **1000**, each location **1002–1006** functions as both a server and a client, and minor latency between the locations is not critical because within each location accounting and physical control remain linked. Latency need be considered only where authority to sell or transfer inventory (objects in the sources **1002–1006**) is separate from the physical control of the inventory.

With the intranet **1000**, the author of an object has physical control over that object and thus may decide what objects are to be exposed for searching by other locations. As a result, the intranet **1000** is well suited for high-security management systems that typically require elaborate security procedures to prevent unauthorized duplication of data. For example, assume there are 200 remote information generators (offices, salespeople, etc.). With the intranet **100**,

data access to information in the objects is maintained through the use of the references available to both the central location and the remote.

The intranet **1000** also provides a more effective means to organize and describe organizational data, creating a much more flexible environment for data retention handling. A data retention handling system has two primary goals: 1) eliminate obsolete data to prevent confusion with current data and reduce storage requirements; and 2) reduce liability. Typically, hierarchical storage management ("HSM") systems have been used for these purposes. A HSM system stores frequently-used or relatively new files on high-speed, immediately available, and most expensive storage media. Older files or files that are not as frequently used are stored on "near-line" storage media that may consist of automatically mounted tape drives or CD-ROMs. Old files or files that are almost never used are stored off-line on tape or other inexpensive high-capacity media. Some files may eventually be deleted if they fall within certain parameters of usage, type, or age. The intranet **1000** overcomes these potential difficulties of a HMS system. For example, in the intranet **1000** duplicate copies of records are not maintained at each location, thereby eliminating the need for hierarchical storage media to provide the required access to stored records.

The agent **204** may also generate ratings for objects stored on the associated sites so that users may filter their searches based upon the generated ratings. For example, in one embodiment, an owner of a web site provides a rating of his site, such as a "G," "R," or "X" rating. In addition, the web host, on which the agent **204** runs, also provides a rating that the host believes applies to the site. The agent **204** then parses the pages on the site and looks for adult content "trigger" words, such as "XXX" or "XXX-Rated." If the agent **204** finds enough occurrences of such trigger words, the agent "flags" the web site for review to determine the correct rating for the site. To rate the site, the agent **204** compares the words in the web pages to words in a list of ratings values. The list of ratings values may be, for example, words that are generally found on adult web sites, such as profane and sexually explicit words. The list of ratings values may be generated by a human or may be automatically generated by the agent **204**. To automatically generate the list, the agent **204** could, for example, parse known adult web sites. Such known adult web sites could be identified by determining those sites in the catalog that include the phrases "adult content" or "X-rated." Once these sites are identified, the agent parses the pages and determines frequently used words on such pages, and may also determine the frequency with which such words occur on these pages. The frequently used words and associated frequencies are then compiled to form the list of ratings values. After flagging web sites for review, the review may be either through human review of the web site or through automated review performed by the agent **204**. In automated review of flagged web sites, the agent **204** could, for example, determine the frequency of occurrence of words in the list of ratings values, and then set the rating of the web site as a function of the frequency. For example, if the frequency is greater than some threshold T1, the web site is rated "R," and if greater than a second threshold T2, where T2>T1, the site is rated "X."

One proposed system for rating web pages on the Internet is described in *A Best Practices Model* by Members of the Information Society Project at Yale Law School, J. M. Balkin, Beth Simone Noveck, Kermit Roosevelt (Jul. 15, 1999), which may be found at http://webserver.law.yale.edu/infosociety/. In this proposed system, three layers are implemented to provide for rating web pages. The first layer includes a basic vocabulary of, for example, thirty to sixty terms that are used in rating a web page by a first party, typically the site owner containing the web page. The second layer includes rating templates developed to reflect a particular ideology. Third parties, such as the NAACP or Christian Coalition, would develop such templates to reflect a particular value system. The templates would include terms in the basic vocabulary being categorized and scalar values assigned to each item to reflect the value system. Finally, in layer three individuals could customize or modify a template to suit their individual values. For example, a template developed by the Christian Coalition could be further modified to include scalar values for web sites designated as racist by the NAACP.

The indexing system **200** could utilize such a rating system to perform filtering of search results at the central server **202**. For example, user's browsers could be registered with the central server **202**, and part of this registration would include selection of a template and any desired modifications to the selected template. Thereafter, whenever the user's browser applies a search query to the central server **202** the browser registration is identified and the search results generated in response to the query are "filtered" according to the template and any template modifications associated with the registered browser.

The indexing system **200** also may perform adult-content locking. In conventional search engines, adult-content web sites are automatically provided in response to applied search queries. The only way for a user to filter adult-content is through a filter on his browser. Thus, current search engines are "opt-in" only in that the search engine does not preclude adult-content pages from being returned in response to applied search queries. Conversely, in one embodiment of the indexing system **200**, the user is automatically opted out of receiving adult-content web pages in response to applied search queries. The user must reverse this default "opt-out" status and elect receive adult-content web pages in the system **200**. This could be done, for example, by registering a browser with the system **200** so that when the registered browser is identified adult-content web sites will be returned in response to applied search queries. Alteratively, a machine level lock using the computer or machine identification, such as the CPU or Windows identification number, could be utilized. In this approach, regardless of the browser being utilized on the computer, adult-content is either returned or not returned in response to applied search queries. This approach may be particularly desirable for parents who want to preclude their children from accessing adult-content since a child cannot merely use a new browser on the same machine and thereby circumvent the filter the parent has on his or her browser.

The indexing system **200** may also perform ranking of web pages having references in the central index. First, the agent **204** may perform positional and contextual rankings for particular words in the web pages on a site. The positional rankings assign a ranking value to a word based upon, for example, the location of the word in the web page and the position of the word relative to other words in the page. The contextual ranking is determined using contextual information about the site contained in the brochure **206**. For example, if a word in a web page corresponds to a category as listed in the brochure **206**, the word will be assigned a higher ranking. In addition to rankings generated by the agent **204**, the central server **202** also generates rankings for the central index. For example, the central server **202** may generate rankings based upon whether a page is a source or

reference to the desired data. Rankings may also be determined based upon user input such as the usage or popularity of a site as measured by how often the site is linked as the source site in other sites, or through positive comments entered by users about the context or ranking of a site. All the methods of ranking just described are know as static rankings, meaning that the ranking is determined before a particular search query is applied.

In addition to static rankings at the central server **202**, the central server may also perform dynamic ranking of search results. Dynamic rankings are a function of the applied search query, and are not predetermined and independent of the query. For example, if the applied search query is "red barn," the word "barn" is probably more important than "red" so search results including the word "barn" will have their ranking increased relative to those containing only the word "red." Furthermore, ratings could be applied to search queries to create another type of dynamic ranking at the central server **202**. Finally, a user may select which ones of the previous methods of rankings should be applied in ranking search results generated in response to his applied query. For example, a user could specify that his search results are to be ranked only on the basis of popularity, or only on the basis of positional and contextual rankings and the applied search query. Another example for the use of dynamic ranking is using the information in the brochure **206**, the search results can be ranked dynamically based on the geographic distance from the searcher.

The server architecture of the system **200** will now be described. The server architecture provides a number of services which support the management and use of index information. The system is divided into several components which can be run on different machines, as needed, in a truly distributed architecture. The design must scale well and be self-healing wherever possible. To make this possible, Jini technology plays an important role in the architecture and services are exposed using that infrastructure. As components are brought online, they advertise that their existence to the local Jini lookup service. This information is automatically propagated to services who need access to other services and handshaking brings elements into the Jini community as they are announced. If non-critical parts of the system become unavailable, the system is able to compensate by distributing load to other machines hosting the necessary services.

A load balancer allows round-robin distribution of incoming traffic to web servers and the agent listener. The web servers provide user services like account registration, agent downloads, brochure management, and search capabilities. The AgentListener is a secure socket listener that manages agent connections. One of the components is a UserAccessService, which controls access to the BrochureService. Users can make queries on the search index. These are handled by the QueryDispatchManager, which delegates subqueries to appropriate IndexSegmentServices. Incoming information from agents is added to the MessageQueueService and popped off by the UpdateManagerService, which coordinates information in the BrochureService to ensure we have the latest updates. Agent-collected changes are added and/or removed in the MasterIndexService.

FIG. **20** shows request/response flow with the direction of arrows. The intent is to make clear who is asking for the execution of respective services. The web server, serving up static and dynamic content through Servlets and Java Server Pages, can communicate with the UserAccessService, BrochureService and the QueryDispatchService, but nothing else. The AgentListener can talk to the UpdateManagerService and the MessageQueueService only. An IndexSegmentService is able to initialize itself by asking from information from the MasterIndexService. Finally, the UpdateManagerService can talk to the BrochureService, MessageQueue service and the MasterIndexService. Its job is to keep the MasterIndexService up to date by processing incoming agent messages.

Because we are using Jini, the order in which services are brought up can determine which other services can operate, but does not restrict that order in any way. If an UpdateManagerService is unavailable, for example, the system will not process updates from the message queue, but processing will resume as soon as the UpdateManagerService is brought up again. As long as more than one instance of a given service is available, the system can discover those services automatically, as they are brought online. An IndexSegmentService is associated with a given IndexSegmentRange, which determines the prefix character range for the index content.

When an IndexSegmentService is brought online, it automatically becomes available to the QueryDispatchService. If one of these services are reinitialized periodically, the update will be completely transparent, so long as other IndexSegmentService cover the same IndexSegmentRange. This might be a single server or may be distributed arbitrarily across a number of IndexSegmentService instances. So long as a QueryDispatchService instance is available to the web servers, and sufficient IndexSegmentService instances are available to cover the full range of possible tokens, the system is capable of executing queries.

The data structures are critical to the correct operation of a complex system. The following description outlines the more important structures that represent the means by which subsystems may interact or store their information persistently in the system **200**.

Persistent information is stored in a database or in temporary files on the system **200**. The database tables relate to each other as shown in FIG. **21**.

The packages presented in FIG. **22** are directly associated with services, components, or conceptual groupings in the system **200**. Major services are represented by their own package, with supporting classes included. Components are given separate packages where applicable. Some services and components accomplish the same tasks and are, naturally, in the same package. Supporting classes, such as database, networking and servlets are grouped into conceptual packages for clarity.

Note that the packages are currently presented in alphabetical order, but may be reorganized in a later revision to reflect the three tiered nature of the architecture of the system **200**. Low level utility packages should be listed first, followed by component/manager packages, Jini service packages, and finally independent applications.

In FIG. **23**, packages are categorized in three ways. They are either low-level utility packages, components, applications and services or user interface elements. Support packages, like the database, catalog, html and xml packages, provide a foundation for other program functionality. A few of the services, the message and index services, for example, are grouped as shared because several of their classes provide functional capabilities between both the agent and server elements. The brochure package is also shared. The application and service level packages construct the agent and the server-side Jini services. Taken together, the classes in these packages function together as a complete, integrated, distributable system.

Referring to FIG. **23**, user interface elements are grouped into the following packages. The com.activeindexing.ui.app package contains classes related to console-based interfaces.

The com.activeindexing.ui.app package contains classes related to web-based user interfaces and contains classes related to application user interfaces.

The agent **204** has its own package as shown in FIG. **24**. The agent **204** has its own package primarily for distribution reasons.

The agent package, com.activeindexing.agent contains classes related to the host agent.

Referring to FIG. **23**, the collection of server of packages provides high level server-side Jini services to the system.

FIG. **25** illustrates the com.activeindexing.server.access package contains, which classes related to the UserAccessService.

The com.activeindexing.server.database package of FIG. **23** contains classes related to database access and record handling and is shown in more detail in FIG. **26**.

Referring to FIG. **23**, the com.activeindexing.server.query package contains classes related to the QueryDispatchService, as shown in more detail in FIG. **27**.

The com.activeindexing.server.servlet package contains classes related to Servlets and web servers, as shown in more detail in FIG. **27**.

The com.activeindexing.server.update package of FIG. **23** contains classes related to the update manager, as shown in more detail in FIG. **28**.

Referring to FIG. **23**, the shared package contains elements which can act as components within the system, used by one or more services or applications.

The com.activeindexing.shared.brochure package is shown in more detail in FIG. **29** and contains classes related to Brochure handling.

The com.activeindexing.shared.index package of FIG. **23** contains classes related to indexing and includes the IndexSegmentService as shown in more detail in FIG. **31**.

The com.activeindexing.shared.message package of FIG. **23** contains classes related to the MessageQueueServiceas shown in more detail in FIG. **32**.

The com.activeindexing.shared.rating package of FIG. **23** contains classes related to rating systems, as shown in more detail in FIG. **33**.

The com.activeindexing.shared.schedule package of FIG. **23** contains classes related to the ScheduleManager, as shown in FIG. **34** in more detail.

The com.activeindexing.shared.signature package of FIG. **23** contains classes related to the file signatures and hash calculations, as shown in more detail in FIG. **34**.

The com.activeindexing.shared.validate package of FIG. **23** contains classes related to field validation, as shown in more detail in FIG. **35**.

Referring to FIG. **23**, the document-related packages, com.activeindexing.doc.html, contains classes related to HTML tokenizing and parsing, as shown in more detail in FIG. **36**.

The com.activeindexing.doc.report package of FIG. **23** contains classes related to reporting, as shown in more detail in FIG. **37**.

The XML package of FIG. **23**, com.activeindexing.doc.xml, contains classes related to XML file management as shown in more detail in FIG. **38**.

The utility package of FIG. **23** contain low-level utility packages which can be used by any other package.

The config package, com.activeindexing.util.config, contains classes related to configuration file handling, as shown in more detail in FIG. **39**.

The I/O package of FIG. **23**, com.activeindexing.util.io, contains utility classes related to input/output operations as shown in more detail in FIG. **40**.

The jini package of FIG. **23**, com.activeindexing.util.jini, contains classes related to Jini services as shown in more detail in FIG. **41**.

The log package of FIG. **23**, com.activeindexing.util.log, contains classes related to the log files, as shown in more detail in FIG. **42**.

The network package of FIG. **23**, com.activeindexing.util.net, contains utility classes related to networking, as shown in more detail in FIG. **43**.

The snmp package of FIG. **23**, com.activeindexing.util.snmp, contains classes related to the Simple Network management Protocol, as shown in more detail in FIG. **44**.

The above description does not include user interface, the XML subsystem, transactions for change requests, or a message format, but one skilled in the art will understand suitable implementations for each of these components.

FIG. **45** is a functional data flow diagram illustrating an alternative embodiment of the central cataloging site of FIG. **2**. In FIG. **45**, a web server **4700** is the main gateway for all agent **204** program update requests, agent program downloads, and search requests. An update batch processor **4702** receives, stores, and applies update batches created by remote agents **204**, and also transmits copies of the batches to redundant remote catalog sites. A remote update batch processor **4704** receives, and applies batches received from a master catalog site to a local index server for the purposes of redundancy. An index server **4706** stores all search index information in a series of database segments, and creates result sets from queries applied to it as a result of search requests received by the web server **4700**.

The system of FIG. **45** includes an agent program storage area **4708** containing copies of agent **204** programs and the digital signatures of those programs for the various host operating systems which use agents to generate web site updates. An update batch storage area **4710** contains the received update batches transmitted by agent programs **204** on remote hosts, and these batches are deleted after processing. An index segment storage area **4712** contains a subset of the total index database for the index server **4706**. For example, a single segment might contain the keyword fields for all of the keywords beginning with the letter "A". Typically, these storage areas will be placed on high-speed RAID storage systems. An index segment storage twin area **4714** is identical to the storage area **4712**. The purpose of the twin area **4714** is to provide access to existing index information while the corresponding index segment storage area is being updated. This permits updates to be applied to a segment without requiring record locking. The index server **4706** is simply notified as to which segment areas are available for search processing. Once updated, the area **4712** or **4714** becomes available again. An index signature storage area **4716** that stores the current digital signature of the index for a particular site serviced by an agent **204** on a remote host.

In operation of the system of FIG. **45**, the agent program, upon starting on a remote host, will query the web server **4700** to determine if the local agent program digital signature matches that of the agent program digital signature stored at the catalog site. If the local agent **204** program

determines that the digital signatures of the agent programs do not match, the agent program will retrieve a new copy of itself from the web servers **4700** and restart itself after performing the appropriate local operations. Before commencing local processing, the agent program **204** checks the digital signature of the existing site index on the catalog site with the digital signature of the site stored locally. If the two signatures match, a differential transmission of catalog information will occur. Otherwise, the entire catalog will be regenerated and transmitted, and the catalog site will be instructed to delete any existing catalog entries for the site. Once a differential or full catalog update has been generated, the agent program **204** contacts the update batch processor **4702** at the catalog site and transmits the contents of the update. Upon receiving confirmation of receipt, the agent program **204** performs clean up and post-processing operations, then suspends itself until the next processing cycle.

The update processor **4702** periodically updates the index segments on the index server **4706**. All updates received are applied as batches to retain data integrity on the index server **4706**. The update processor **4702** separates update information as required to match the segments on the index server **4706**, then updates each segment storage area **4712** and each segment storage twin area **4714**. While a segment storage area **4712**, **4714** is being updated, its counterpart is available for search request processing. Once all updates have been applied, the digital signature of the index for the site is updated in the index signature storage area **4716** and the batch is deleted from the update batch storage area **4710**.

In processing search requests, the web servers **4700** receive and interpret the search requests from remote portals or web browsers. Each search request is preprocessed to divide the request into sub-requests as required for each index segment, then the index server **4706** is requested to perform search queries on each relevant segment. More than one index segment may be queried simultaneously. The index server **4706** determines which index segment storage areas **4712**, **4714** are available for use, applies the search request, and transmits the results to the web server **4700** which, in turn, collects and collates all search results and transmits these results back to the requesting system in a formatted manner.

According to another embodiment of the agent **204**, the agent calculates a value representing the distance and text between objects and thereby determines which objects at a site are most likely to relate to each other. At the catalog site, these relationship values are combined with the relationship values from other sites to create a relationship value table. This relationship value table represents the likelihood of an object occurring together with another object. This table may be used to refine searches and create relevance ranking.

It is to be understood that even though various embodiments and advantages of the present invention have been set forth in the foregoing description, the above disclosure is illustrative only, and changes may be made in detail, and yet remain within the broad principles of the invention.

Therefore, the present invention is to be limited only by the appended claims.

We claim:

1. A computer-based method for exchanging files between peer computers, the method comprising:

transmitting a request for a file from a first peer computer to a server computer through a network;

processing the file request at the server computer by searching an index of information to determine a second peer computer wherein the requested file is stored; and

transmitting an instruction to the second peer computer directing the second peer computer to transmit the file to the first peer computer.

2. A computer-readable medium having computer-executable instructions operable for exchanging files between peer computers, the computer-executable instructions operable for:

transmitting a request for a file from a first peer computer to a server computer through a network;

processing the file request at the server computer by searching an index of information to determine a second peer computer wherein the requested file is stored; and

transmitting an instruction to the second peer computer directing the second peer computer to transmit the file to the first peer computer.

3. The computer-readable medium of claim **2** further comprising computer-executable instructions operable for transmitting the network location of the second peer computer to the first peer computer.

4. The computer-readable medium of claim **2** wherein the network locations of the first peer computer and the second peer computer are anonymous.

5. The computer-readable medium of claim **2** wherein the server computer is also one of the peer computers.

6. The computer-readable medium of claim **2** further comprising computer-executable instructions operable for sending a notification to the first peer computer that the requested file has been located.

7. The computer-readable medium of claim **6** wherein the notification is sent to a pre-determined email address associated with the first peer computer.

8. The computer-readable medium of claim **2** further comprising computer-executable instructions operable for sending a notification to the second peer computer that the located file as been requested.

9. The computer-readable medium of claim **8** wherein the notification is sent to a pre-determined email address associated with the second peer computer.

* * * * *